LECTURE NOTES IN SEMI-TENSOR PRODUCT OF MATRICES

ANALYSIS AND CONTROL OF LOGICAL DYNAMIC SYSTEMS

矩阵半张量积讲义



逻辑动态系统的分析与控制

程代展　齐洪胜 / 著

科学出版社

作 者 简 介

程代展　1970 年毕业于清华大学, 1981 年于中国科学院大学获硕士学位, 1985 年于美国华盛顿大学获博士学位. 从 1990 年起, 任中国科学院系统科学研究所研究员. 曾经担任国际自动控制联合会 (International Federation of Automatic Control, IFAC) 理事 (Council Member), IEEE 控制系统协会 (Control Systems Society, CSS) 执委 (Member of Board of Governors), 中国自动化学会控制理论专业委员会主任, IEEE CSS 北京分会主席等, 国际期刊 *Int. J. Math Sys., Est. Contr.* (1991—1993), *Automatica* (1998—2002), *Asia. J. Control.* (1999—2004) 的编委, *International Journal on Robust and Nonlinear Control* 的主题编委, 国内期刊 *J. Control Theory and Application* 主编, 《控制与决策》的副主编及多家学术刊物的编辑. 已经出版了 16 本论著, 发表了 270 多篇期刊论文和 150 多篇会议论文. 他的研究方向包括非线性控制系统、数值方法、复杂系统、布尔网络控制、基于博弈的控制等. 曾两次作为第一完成人获国家自然科学奖二等奖 (2008, 2014), 中国科学院个人杰出科技成就奖 (金质奖章, 2015), 其他省部级一等奖两次、二等奖四次、三等奖一次. 2011 年获国际自动控制联合会 *Automatica* (2008—2010) 的最佳论文奖. 2006 年入选 IEEE Fellow, 2008 年入选 IFAC Fellow.

程代展是矩阵半张量积理论的首创人.

齐洪胜　中国科学院数学与系统科学研究院副研究员, 中国科学院大学岗位教授. 2008 年于中国科学院数学与系统科学研究院获博士学位. 2008 年 7 月至 2010 年 6 月在中国科学院系统控制重点实验室从事博士后研究工作. 已经出版了 4 本著作, 发表了 60 多篇期刊论文和会议论文. 主要研究兴趣包括逻辑动态系统、博弈与控制、量子网络等. 2011 年获国际自动控制联合会 *Automatica* (2008—2010) 理论/方法类最佳论文奖, 2014 年获国家自然科学奖二等奖 (排名第二).

矩阵半张量积讲义

卷二：逻辑动态系统的分析与控制

程代展　齐洪胜　著

科　学　出　版　社

北　京

内 容 简 介

矩阵半张量积是近二十年发展起来的一种新的矩阵理论. 经典矩阵理论的最大弱点是其维数局限, 这极大地限制了矩阵方法的应用. 矩阵半张量积是经典矩阵理论的发展, 它克服了经典矩阵理论对维数的限制, 因此, 被称为跨越维数的矩阵理论. 《矩阵半张量积讲义》的目的是对矩阵半张量积理论与应用做一个基础而全面的介绍. 计划出五卷. 卷一: 基本理论与多线性运算; 卷二: 逻辑动态系统的分析与控制; 卷三: 有限博弈的矩阵方法; 卷四: 泛维数动力系统; 卷五: 矩阵半张量积的其他应用. 《矩阵半张量积讲义》是对这个快速发展的学科分支做一个阶段性的小结, 以期为其进一步发展及应用提供一个规范化的基础.

本书是《矩阵半张量积讲义》的第二卷. 本书所需要的预备知识仅为大学本科工科专业的数学知识, 包括线性代数、微积分、常微分方程、初等概率论. 相关的线性系统理论及点集拓扑、抽象代数、微分几何等的初步概念已在卷一附录中给出.读者亦可略过相关部分, 这些不会影响对本书基本内容的理解.

本书可供离散数学、自动控制、计算机、系统生物学、博弈论及相关专业的高年级本科生、硕士研究生、博士研究生、青年教师及科研人员使用.

图书在版编目(CIP)数据

矩阵半张量积讲义. 卷二, 逻辑动态系统的分析与控制/程代展, 齐洪胜著. —北京: 科学出版社, 2022.2

ISBN 978-7-03-070660-7

Ⅰ. ①矩… Ⅱ. ①程… ②齐… Ⅲ. ①矩阵-乘法 Ⅳ. ①O151.21

中国版本图书馆 CIP 数据核字 (2021) 第 233108 号

责任编辑: 李 欣 李 萍 / 责任校对: 彭珍珍
责任印制: 吴兆东 / 封面设计: 无极书装

科 学 出 版 社 出版
北京东黄城根北街 16 号
邮政编码: 100717
http://www.sciencep.com

北京虎彩文化传播有限公司 印刷
科学出版社发行 各地新华书店经销
*

2022 年 2 月第 一 版 开本: 720 × 1000 1/16
2022 年 2 月第一次印刷 印张: 23 3/4
字数: 479 000

定价: 188.00 元

(如有印装质量问题, 我社负责调换)

前　　言

矩阵理论是被公认起源于中国的一个数学分支. 美国哥伦比亚特区大学教授 Katz 在著名数学史著作[72]中指出: “The idea of a matrix has a long history, dated at least from its use by Chinese scholars of the Han period for solving systems of linear equations.” (矩阵的思想历史悠久, 它的使用至少可追溯到汉朝, 中国学者用它来解线性方程组.) 英国学者 Crilly 的书[8]中也提到: 矩阵起源于 “公元前 200 年, 中国数学家使用了数字阵列”. 矩阵理论是这两本书中唯一提到的始于中国的数学分支, 大概确实是仅见的.

从开始不甚清晰的思考到如今形成一个较完整的体系, 矩阵半张量积走过了大约二十个年头. 开始, 人们质疑它的合理性, 有人提到: “华罗庚先生说过, 将矩阵乘法推广到一般情况没有意义.” 后来, 又有人质疑它的原创性, 说: “这么简单的东西怎么会没有前人提出或讨论过?” 到如今, 它已经被越来越多的国内外学者所肯定和采用.

回顾矩阵半张量积的历史, 催生它的有以下几个因素:

(1) 将矩阵乘法与数乘相比, 矩阵乘法的两个明显的弱点是: ① 维数限制, 只有当前因子的列数与后因子的行数相等时, 这两个矩阵才可相乘; ② 无交换性, 一般地说, 即使 AB 和 BA 都有定义, 但 $AB \neq BA$. 因此, 将普通矩阵乘法推广到任意两个矩阵, 并且让矩阵乘法具有某种程度的交换性, 将会大大扩大矩阵方法的应用.

(2) 将矩阵加法与数加相比, 虽然矩阵加法也可以交换, 但其实维数的限制更为苛刻. 即行、列两个自由度都必须相等. 有没有办法, 让不同维数的矩阵也能相加? 而这个加法, 必须有物理意义而且有用.

(3) 经典的矩阵理论其实只能处理线性函数 (线性方程) 或双线性函数 (二次型). 如果是三阶或更高阶的多线性函数, 譬如张量, 还能用矩阵方法表示并计算它们吗? 当然, 如果矩阵方法能用于处理更一般的非线性函数, 那就更好了.

上述这些问题曾被许多人视为矩阵理论几乎无法逾越的障碍. 然而, 让人们吃惊的是, 矩阵半张量积几乎完美地解决了上述这些问题, 从而催生了一套新的矩阵理论, 被我们称为跨越维数的矩阵理论.

目前, 它已经被应用于许多领域, 包括:

(1) 生物系统与生命科学. 这个方面目前的一些进展包括: 文献 [142] 研究

了 T 细胞受体布尔控制网络模型, 给出了寻找它所有吸引子的有效算法; 关于大肠杆菌乳糖操纵子网络稳定与镇定控制的设计, 文献 [88,89] 分别给出了不同的设计方法, 证明了方法的有效性; 对黑色素瘤转移控制, 文献 [33] 给出了最优控制的设计与算法; 基因各表现型之间的转移控制, 文献 [50, 51] 给出了转移表现型的估计并精确地给出了最短控制序列; 等等.

将布尔网络控制理论用于生物系统是一个非常有希望的交叉方向. 进一步的研究需要跨学科的合作.

(2) 博弈论. 有限博弈本质上也是一个逻辑系统. 因此, 矩阵半张量积是研究有限博弈的一个有效工具. 目前, 矩阵半张量积在博弈论中的一些应用包括: 网络演化博弈的建模和分析[32, 55]; 最优策略与纳什均衡的探索[137]; 有限势博弈的检验与势函数计算[31], 网络演化博弈的演化策略及其稳定性[30]; 有限博弈的向量空间结构[34, 59]; 等等.

(3) 图论与队形控制. 这方面的代表性工作包括: 图形着色及其在多自主体控制中的应用[117]; 队形控制的有限值逻辑动态系统表示[134]; 对超图着色及其在存储问题中的应用[100]; 图形着色的稳健性及其在时间排序中的应用[128]; 等等.

(4) 线路设计与故障检测. 这一方面的一些现有研究工作包括: k 值逻辑函数的分解; 隐函数存在定理[29]; 故障检测的矩阵半张量积方法[6, 87, 96]; 等等.

(5) 模糊控制. 在模糊控制方面的一些初步工作包括: 模糊关系方程的统一解法[28]; 带有耦合输入和/或耦合输出的模糊系统控制[43]; 对二型模糊关系方程的表述和求解[129]; 空调系统的模糊控制器设计[124]; 等等.

(6) 有限自动机与符号动力学. 这方面的部分工作包括: 有限自动机的代数状态空间表示与可达性[125]; 并应用于语言识别[130]; 有限自动机的模型匹配[126]; 有限自动机的可观性与观测器设计[127]; 布尔网络的符号动力学方法[61]; 有限自动机的能控性和可镇定性[131]; 等等.

(7) 编码理论与算法实现. 这方面的一些研究包括: 对布尔函数微分计算的研究[138]; 布尔函数的神经网络实现[137]; 非线性编码[97, 139, 147, 149]; 等等.

(8) 工程应用. 代表性工作包括: 电力系统[5]; 在并行混合电动汽车控制中的应用[4, 121]; 等等.

前面所列举的仅为矩阵半张量积理论及其应用研究中的极少一部分相关论文, 难免以偏概全. 在一群中国学者的主导和努力下, 矩阵半张量积正在发展成为一个极具生命力的新学科方向. 同时, 它也吸引了国际上许多学者的重视和加入. 目前, 用矩阵半张量积为主要工具的论文作者, 除中国外, 有意大利、以色列、美国、英国、日本、南非、瑞典、新加坡、德国、俄罗斯、澳大利亚、匈牙利、伊朗、沙特阿拉伯等. 矩阵半张量积可望成为当代中国学者对矩阵理论的一个重要贡献.

有关矩阵半张量积的书, 算起来也已经有好几本了. 这几本书各有特色. 例如: 文献 [1], 这本书写得比较早, 对矩阵半张量积的普及和推广起到了一定的作用, 但当时矩阵半张量积理论还很不成熟, 所以显得很粗糙, 虽然后来出了第 2 版, 但仍然改进不大; 文献 [27], 它力图包括更多的应用, 对工程人员可能有较大帮助, 但是对矩阵理论本身缺乏系统疏理, 不便系统学习; 文献 [2], 它强调用半张量积方法统一处理逻辑系统、多值逻辑系统及有限博弈等, 对矩阵半张量积理论自身的讨论不多; 文献 [93], 该书是一本新书, 它对某些控制问题进行了较详尽的剖析, 这是它的贡献, 但它缺少对矩阵半张量积理论全局的把控; 文献 [36], 它是大学本科教材, 内容清晰易懂, 但作为科研参考书显然是不够的; 其他如文献 [22], 它专门讨论布尔网络的控制问题; 文献 [5], 它只关心电力系统的优化控制问题; 文献 [38], 它主要考虑泛维系统的建模与控制. 因此, 已有的关于矩阵半张量积的论著, 内容或已过时, 或过于偏重部分内容.

这套书定名为《矩阵半张量积讲义》, 计划出五卷. 卷一: 基本理论与多线性运算; 卷二: 逻辑动态系统的分析与控制; 卷三: 有限博弈的矩阵方法; 卷四: 泛维数动力系统; 卷五: 矩阵半张量积的其他应用.

由于矩阵半张量积理论与方法发展过快, 许多理论结果、计算公式以及综合和归纳方法等被其后的新成果代替. 这给初学者和科研人员均带来了一定的不便. 本套丛书的目的, 是为矩阵半张量积理论提供一个至今尽可能完整和先进的理论框架, 让它体系完善、结构清晰、公式简洁. 同时, 将矩阵半张量积的主要应用进行详细分析, 使其原理准确易懂, 方法明确有效, 便于读者不走弯路, 迅速到达学科前沿. 同时, 内容尽可能增加启发性, 讲清来龙去脉, 给出详尽证明, 以便读者举一反三, 应用自如. 总之, 希望本丛书为读者搭建一个工作平台, 提供一个基准、一块进一步学习、应用及发展矩阵半张量积的奠基石.

本书是第二卷, 主要讨论矩阵半张量积方法在逻辑动态系统中的应用, 特别注重布尔网络的矩阵半张量积表示、结构分析以及布尔控制网络的相关控制问题的解. 本卷共 12 章. 第 1 章介绍布尔函数的基本性质及其表示方法; 第 2 章介绍逻辑函数及静态逻辑网络. 这两章是全书的基础. 第 3 章讨论逻辑动态系统的拓扑结构, 重点是不动点与极限环的计算、吸引域的确定等. 第 4 章与第 5 章分别讨论逻辑控制系统的能控性与能观性, 这是控制系统最基本的两个性质. 第 6 章介绍代数状态空间方法, 包括坐标变换、正规子空间及控制不变子空间等. 第 7 章讨论解耦问题, 包括干扰解耦、状态空间解耦以及输入输出解耦等. 第 8 章探讨布尔网络的稳定性与布尔控制网络的可镇定性. 第 9 章讨论辨识问题, 包括布尔网络的辨识与布尔控别网络的辨识. 第 10 章讨论逻辑系统的输出调节问题, 包括输出跟踪的可解性及跟踪控制的设计. 第 11 章讨论概率布尔网络, 包括其代数状态空间的表达、能控性、稳定性与镇定、能观测性等. 第 12 章研究最优控制问题,

包括时间最优、平均收益最优、加权总和最优等.

这套书只要求读者具有大学本科生工科专业所需掌握的数学工具, 但部分内容涉及一些近代数学的初步知识. 为了使本丛书具有良好的完备性, 以增加可读性, 第一卷书末添加了一个附录, 对一些用到的近代数学知识做了简要介绍. 如果仅为阅读本丛书, 这些知识也就足够了.

逻辑动态系统及其控制是矩阵半张量积理论中成果最多的, 也是最活跃、发展最快的一部分, 由于篇幅及囿于笔者的学识, 讨论到的内容难免以偏概全, 挂一漏万. 同时, 还有许多有意义的结果未能涉及. 例如, 为降低计算复杂性而给出的算法设计[142,143], 基于有限自动机的广义布尔网络[145] 等. 笔者才疏学浅, 疏漏错误难免, 敬请读者以及有关专家不吝赐教.

程代展　齐洪胜
于中国科学院数学与系统科学研究院
2020 年 10 月

目　　录

数 学 符 号

$\mathbb{C}$	复数集
$\mathbb{R}$	实数集
$\mathbb{Q}$	有理数集
$\mathbb{Q}_+$	正有理数集
$\mathbb{N}$	自然数 (正整数) 集
$:=$	“定义为 ······”
$\mathcal{M}_{m\times n}$	$m\times n$ 矩阵集合
$\mathbf{1}_k$	$[\underbrace{1,1,\cdots,1}_{k}]^{\mathrm{T}}$
$\mathbf{0}_k$	$[\underbrace{0,0,\cdots,0}_{k}]^{\mathrm{T}}$
δ_n^k	单位矩阵 I_n 的第 k 列
δ_n^0	$\delta_n^0=\mathbf{0}_n$
$m\|n$	m 为 n 的因子
$\mathcal{L}_{m\times n}$	$m\times n$ 维逻辑矩阵集合
$\varUpsilon_n$	n 维概率向量集合
$\varUpsilon_{m\times n}$	$m\times n$ 维概率矩阵集合
$\delta_k[i_1,\cdots,i_s]$	一个逻辑矩阵, 其第 j 列为 $\delta_k^{i_j}$
$\delta_k\{i_1,\cdots,i_s\}$	$\left\{\delta_k^{i_1},\cdots,\delta_k^{i_s}\right\}\subset\varDelta_k$
$\mathcal{B}_{m\times n}$	$m\times n$ 维布尔矩阵集合
$\otimes$	矩阵的 Kronecker 积 (张量积)
$*$	矩阵的 Khatri-Rao 积
$V_c(A)$	矩阵 A 的列堆式
$V_r(A)$	矩阵 A 的行堆式
$\ltimes$	一型矩阵-矩阵左半张量积
$\rtimes$	一型矩阵-矩阵右半张量积
$\mathcal{D}$	集合 $\{T,F\}$ 或 $\{1,0\}$
$\mathcal{D}_k$	集合 $\left\{0,\dfrac{1}{k-1},\cdots,\dfrac{k-2}{k-1},1\right\}$ 或 $\{1,2,\cdots,k\}$
$\mathrm{Col}(A)$	矩阵 A 的列向量集合

$\mathrm{Col}_i(A)$	矩阵 A 的第 i 个列向量
$\mathrm{Row}(A)$	矩阵 A 的行向量集合
$\mathrm{Row}_i(A)$	矩阵 A 的第 i 个行向量
$\varDelta_k$	$\varDelta_k = \mathrm{Col}(I_k)$
$\varDelta_k^0$	$\varDelta_k^0 = \varDelta_k \cup \{\delta_k^0\}$
$\mathrm{tr}(A)$	矩阵 A 的迹
$\sigma(A)$	矩阵 A 的特征值集合
$\mathbf{S}_k$	k 阶对称群
$\mathbf{A}_k$	k 阶交错群
$W_{[m,n]}$	(m,n) 阶换位矩阵
$W_{[n]}$	$W_{[n]} = W_{[n,n]}$
$+_{\mathcal{B}}$	布尔加
$\sum\limits_{\mathcal{B}}$	布尔连加
$\times_{\mathcal{B}}$	布尔积
$A^{(k)}$	A 的布尔幂
$\mathrm{span}\,(\cdots)$	由 $\cdots\cdots$ 张成的向量空间
$H < G$	H 为 G 的子群
$H \lhd G$	H 为 G 的正规子群
$\cong$	群同构
$\mathbf{A}$	代数
$\mathrm{GL}(n,\mathbb{R})$	n 阶一般线性群
$\mathrm{gl}(n,\mathbb{R})$	n 阶一般线性代数
$\neg$	(逻辑算子) 非
$\vee$	(逻辑算子) 析取
$\wedge$	(逻辑算子) 合取
$\rightarrow$	(逻辑算子) 蕴涵
$\leftrightarrow$	(逻辑算子) 等价
$\bar{\vee}$	(逻辑算子) 异或
$\uparrow$	(逻辑算子) 与非
$\downarrow$	(逻辑算子) 或非
M_n	非的结构矩阵
M_d	析取的结构矩阵
M_c	合取的结构矩阵
M_i	蕴涵的结构矩阵

M_e	等价的结构矩阵
M_p	异或的结构矩阵
PR_k	降阶矩阵
$a \vee b$ (或 $\mathrm{lcm}(a,b)$)	a, b 的最小公倍数
$a \wedge b$ (或 $\mathrm{gcd}(a,b)$)	a, b 的最大公因数
$\mathbb{Z}_p$	伽罗瓦域
RV_s	s 阶反序算子
MV_s	s 阶反序矩阵
$B(i)$, $i \in \mathbb{Z}_+$	i 的二进制表达式
$\mathcal{B}_n^{\mathcal{F}}$	n 元逻辑函数
$\vec{X}$	逻辑变量 X 的向量表达式

第 1 章　布尔函数的表示与结构分析

布尔函数是逻辑系统的基本表示形式, 因此, 也是本书的主要讨论对象, 研究它们的基本性质对逻辑系统的分析与控制的讨论至关重要. 除了逻辑系统, 布尔函数在编码等中也极其重要, 因此, 布尔函数有多种不同的表达形式. 在不同形式下它们反映出不同的性质. 本章的目的, 是探讨布尔函数的不同表达形式、结构与基本性质.

逻辑矩阵是布尔函数的代数状态空间表示的基本形式, 因此, 也是研究布尔函数性质的基本工具.

逻辑函数指的是自变量与函数值均取自有限集的情况, 包括布尔函数 (即二值逻辑函数)、k 值逻辑函数及混合值逻辑函数. 布尔函数是基础, 因为它们不仅最简单、应用最广, 而且, 由于它们被研究的时间最久, 已知的结论也最多. 同时, 一般逻辑函数的表达和性质, 常可通过与布尔函数类似的方法处理. 布尔函数在计算机科学[119]、密码学[7,17] 等领域中都有大量应用, 本章以矩阵半张量积为工具, 讨论布尔函数与本书相关的一些主要性质.

1.1　布尔函数的代数表示

1.1.1　伽罗瓦域上的逻辑函数

定义 1.1.1　设 $p>1$ 为一素数, 令

$$\mathbb{Z}_p=\{0,1,\cdots,p-1\}.$$

在 $\mathbb{Z}_p$ 上定义加法和乘法如下:

$$\begin{cases} a+_p b:=a+b \pmod p, \\ a\times_p b:=ab \pmod p. \end{cases} \tag{1.1.1}$$

那么, $(\mathbb{Z}_p,+_p,\times_p)$ 就是一个域, 这种域称为伽罗瓦域.

当 $p=2$ 时 $\mathbb{Z}_2=\mathcal{D}$. 为方便计, 我们将 $\mathbb{Z}_2$ 称为布尔域.

用 $\mathcal{B}_n^{\mathcal{F}}$ 表示 n 个自变量的逻辑函数, $P(\mathbb{Z}_2)$ 表示多项式布尔函数. 因为 $\mathbb{Z}_2=\mathcal{D}$, 那么, $\mathbb{Z}_2$ 上的运算能代替逻辑运算吗? 这是可以的.

命题 1.1.1　$\{+_2, \times_2\}$ 是一个完备集, 因此, 任何一个布尔函数都可以用 $\{+_2, \times_2\}$ 表示出来. 换言之, 任何布尔函数均可用 $\mathbb{Z}_2$ 上的多项式表示.

证明　只要一个完备集的每个算子能用多项式表示即可. 考虑 $\{\wedge, \neg\}$, 它是经典逻辑的一个完备集. 所以只要它的元素能用 $\{+_2, \times_2\}$ 表示, 则 $\{+_2, \times_2\}$ 就是一个完备集. 用 F_σ 表示逻辑函数 σ 的多项式形式:

$$\begin{aligned} &F_\wedge(X,Y) = X \wedge Y = X \times_2 Y, \\ &F_\neg(X) = \neg X = X +_2 1, \quad X, Y \in \mathcal{D}. \end{aligned} \tag{1.1.2}$$

□

记 $P(\mathbb{Z}_p)$ 为 $\mathbb{Z}_p$ 上的多项式. 那么, 其上的加、乘为 $+_p$, $\times_p$. 为方便计, 将其简记为 $+$, $\times$ (或省去 $\times$, 即 $X \times_p Y = X \times Y = XY$).

例 1.1.1　设 $F \in P(\mathbb{Z}_2)$.

(i) 直接计算可得

$$\begin{aligned} F_\neg X &= X + 1, \\ F_\wedge(X,Y) &= XY, \\ F_\vee(X,Y) &= XY + X + Y, \\ F_{\bar\vee}(X,Y) &= X + Y, \\ F_\rightarrow(X,Y) &= XY + X + 1, \\ F_\leftrightarrow(X,Y) &= X + Y + 1, \quad X, Y \in \mathcal{D}. \end{aligned} \tag{1.1.3}$$

(ii) 令 $\varphi(X,Y,Z) = (X \vee Y) \leftrightarrow (X \wedge Z)$. 利用 (1.1.3) 可得

$$\begin{aligned} F_\varphi(X,Y,Z) &= (X \vee Y) + (X \wedge Z) + 1 \\ &= (XY + X + Y) + (XZ) + 1 \\ &= XY + XZ + X + Y + 1. \end{aligned}$$

给定一个 $X = \{X_1, X_2, \cdots, X_n\} \in \mathbb{Z}_2^n$. 我们给 X 几个等价的表达形式:

(i) 逻辑形式

$$X = (X_1, X_2, \cdots, X_n), \quad X_i \in \mathbb{Z}_2, \quad i = 1, \cdots, n. \tag{1.1.4}$$

(ii) 向量形式

$$x_i = \vec{X}_i := \begin{bmatrix} X_i \\ 1 - X_i \end{bmatrix}. \tag{1.1.5}$$

$$x = \vec{X} := \ltimes_{i=1}^n x_i. \tag{1.1.6}$$

(iii) 自然数形式

$$d = D(X) := \sum_{i=1}^{n} X_i 2^{n-i}. \tag{1.1.7}$$

注 1.1.1 逻辑变量 $X = \{X_1, X_2, \cdots, X_n\} \in \mathbb{Z}_2^n$ 的几种表达形式的等价性可从以下的转换式得到. 记 $x_i = \vec{X}_i$, $x = \vec{X}$, $d = D(X)$.

(i) 逻辑形式 $\Rightarrow$ 向量形式: 见 (1.1.5)—(1.1.6).

(ii) 逻辑形式 $\Rightarrow$ 自然数形式: 见 (1.1.7).

(iii) 向量形式 $\Rightarrow$ 逻辑形式:

$$x_i = \left(\mathbf{1}_{2^{i-1}}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-i}}^{t}\right) x. \tag{1.1.8}$$

再利用 (1.1.5) 即得.

(iv) 向量形式 $\Rightarrow$ 自然数形式: 设 $x = \vec{X} = \delta_{2^n}^t$, 则

$$d = D(x) = 2^n - t. \tag{1.1.9}$$

(v) 自然数形式 $\Rightarrow$ 逻辑形式: 将 $d = D(X)$ 表示成二进制形式, 记为 $B(d)$. (如果 $B(d)$ 只有 $s < n$ 位, 则在前面补 $n - s$ 个零, 仍记作 $B(d)$) 则

$$B(d) = (X_1, X_2, \cdots, X_n). \tag{1.1.10}$$

(因此, 逻辑形式也可称为二进制形式.)

(vi) 自然数形式 $\Rightarrow$ 向量形式: 设 $d = D(X)$, $x = \vec{X}$, 则

$$x = \delta_{2^n}^{2^n - d}. \tag{1.1.11}$$

下面给一个转换的例子:

例 1.1.2 设 $X = (X_1, X_2, \cdots, X_8) \in \mathcal{D}^8$. $x = \vec{X}$, $d = D(X)$.

(i) 设 $d = 51$, 那么

$$X = B(51) = (0, 0, 1, 1, 0, 0, 1, 1).$$

即 $X_1 = 0$, $X_2 = 0$, $X_3 = 0$, $X_4 = 1$, $X_5 = 0$, $X_6 = 0$, $X_7 = 1$, $X_8 = 1$.

$$x = \delta_{2^8}^{2^8 - 51} = \delta_{256}^{205}.$$

(ii) 设 $X = (1, 1, 0, 0, 1, 0, 1, 0)$, 那么

$$d = 2^7 + 2^6 + 2^3 + 2^1 = 202.$$

$$x = \delta_{2^8}^{2^8-202} = \delta_{256}^{54}.$$

(iii) 设 $x = \vec{X} = \delta_{2^8}^{120}$, 那么

$$d = D(\chi) = 2^8 - 120 = 136,$$
$$X = B(d) = (1,0,0,0,1,0,0,0).$$

给定一个布尔函数 $F : \mathcal{D}^n \to \mathcal{D}$, 记该函数为 $F(X_1, X_2, \cdots, X_n)$. 通常它是逻辑函数表达式. 于是, 将它转化为其他表达式以便将逻辑运算转化为易于计算的代数运算就成了一个合理选择.

(i) 向量表达式: 令 $x_i = \vec{X}_i$, 则存在一个逻辑矩阵 $M_F \in \mathcal{L}_{2\times 2^n}$ 使得

$$f(x) := \vec{F}(x) = M_f \ltimes_{i=1}^n x_i = M_f x.$$

这是本书的主要方法.

(ii) 序列表示法: 把 X 用 $d = D(X)$ 表示, 则 $d = 0, 1, \cdots, 2^n - 1$. F 的序列表示法, 记作 $\mathbf{F}$, 定义为

$$\mathbf{F} = (F(0), F(1), \cdots, F(2^n - 1)). \tag{1.1.12}$$

$\mathbf{F}$ 也称为 F 的序列向量.

这两种表示都是 F 的完整表述, 因此, 它们必然是等价的. 下面验证这种等价关系.

定义 1.1.2　设 $X = (X_1, X_2, \cdots, X_s) \in \mathbb{R}^s$ 为一行向量, 则 s 阶反序算子记作 RV_s, 定义为

$$RV_s(X) = (V_s, X_{s-1}, \cdots, X_1). \tag{1.1.13}$$

注 1.1.2　(i) 设 $Y = (Y_1, Y_2, \cdots, Y_s)^{\mathrm{T}}$ 为一列向量, 则定义

$$RV_s(Y) = (Y_s, Y_{s-1}, \cdots, Y_1)^{\mathrm{T}}. \tag{1.1.14}$$

(ii)

$$RV_s^2(X) = X, \quad \forall X \in \mathbb{R}^s.$$

(iii) 定义反序矩阵, 记作 MV_s, 如下:

$$MV_s = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathcal{M}_{s\times s}. \tag{1.1.15}$$

(iv) 设 $X \in \mathbb{R}^s$ 为一行向量, 则

$$RV_s(X) = XMV_s.$$

设 $Y \in \mathbb{R}^s$ 为一列向量, 则

$$RV_s(Y) = MV_sY.$$

定义 1.1.3 设布尔函数 F 的结构矩阵为 M_F, 则

$$r_F := \mathrm{Row}_1(M_F) \tag{1.1.16}$$

称为 f 的结构向量.

由定义可知如下命题.

命题 1.1.2 设布尔函数 F 的序列向量为 $\mathbf{F}$, 结构向量为 r_F, 那么

$$\mathbf{F} = RV(r_F). \tag{1.1.17}$$

于是, 不同表达式的转化就变得很简单了.

例 1.1.3 (i) 设 $M_F = \delta_2[1,1,2,1,2,1,1,2]$, 那么

$$\mathbf{F} = (0,1,1,0,1,0,1,1).$$

(ii) 设 $\mathbf{F} = (1,1,0,1,1,0,1,0)$, 那么

$$M_F = \delta_2[2,1,2,1,1,2,1,1].$$

1.1.2 布尔函数的多项式表示

对应于布尔变量的三种表达: 逻辑表达、向量表达及自然数表达, 我们也可以得到布尔函数的三种表达: 逻辑表达、向量表达 (代数状态空间表达式) 及多项式表达. 对于前两种表达方法及其转换, 我们已经非常熟悉了. 下面讨论, 如何得到布尔函数的多项式表达.

先讨论如何从逻辑表达式得到多项式表达式.

实际上, 例 1.1.1 已经讨论过这个问题了, 这里将它小结成一个算法.

算法 1.1.1 第一步:

$$F_\neg(X) = X + 1; \quad F_\wedge(X,Y) = XY.$$

由于 $(\neg, \wedge)$ 是完备集, 任何逻辑算子都可用它们表示. 于是, 常用逻辑算子的多项式表达可知. 例如, 见 (1.1.3).

第二步: 一个一般的逻辑表达式都是由常见逻辑表达式复合而成的. 构造常见逻辑表达式的多项式形式的复合函数, 即可得一般逻辑表达式的多项式表示. 例 1.1.1 (ii) 就给出了这样的例子.

第三步: 将得到的复合多项式化简, 注意到 $X+X=0$ 及 $XX=X$, 就可以合并同类项和降次.

由上述算法可得如下结论:

命题 1.1.3　设 $F:\mathbb{Z}_2^n\to\mathbb{Z}^2$, $X=\{X_1,X_2,\cdots,X_n\}\in\mathbb{Z}_2^n$, 那么, F 有如下多项式表达式:

$$
\begin{aligned}
F(X) &= a_0 + a_1X_1 + \cdots + a_nX_n + a_{12}X_1X_2 + \cdots \\
&\quad + a_{n-1,n}X_{n-1}X_n + \cdots + a_{12\cdots n}X_1X_2\cdots X_n \\
&= a_0 + \sum_{k=1}^{n}\sum_{1\leqslant j_1<\cdots<j_k\leqslant n} a_{j_1\cdots j_k}X_{j_1}X_{j_2}\cdots X_{j_k}.
\end{aligned} \tag{1.1.18}
$$

再讨论如何从自然数表达式得到多项式表达式.

定义两个记号:

(i) 设 $X,a\in\mathcal{D}$, 定义

$$
X^a := \begin{cases} X, & a=1, \\ X+1(\text{或 } \neg X), & a=0. \end{cases} \tag{1.1.19}
$$

(ii) 设 $X,c\in\mathcal{D}^n$, 定义

$$
X^c := \prod_{i=1}^{n} X_i^{c_i}. \tag{1.1.20}
$$

利用这些记号, 可以得到从函数的自然数表达式到多项式表达式的转换公式.

命题 1.1.4　设布尔函数 $F:\mathcal{D}^n\to\mathcal{D}$ 的自然数表达式为 $\mathbf{F}=(F(0),F(1),\cdots,F(2^n-1))$, 则

$$
F(X) = \sum_{i=1}^{2^n-1} F(i)X^{B(i)}, \tag{1.1.21}
$$

这里, $B(i)\in\mathcal{D}^n$ 是 i 的二进制表达式.

例 1.1.4　考虑

$$
F(X_1,X_2,X_3) = (X_1\wedge X_2)\leftrightarrow X_3,
$$

那么, 我们有

$$
\begin{aligned}
&F(0)=F(0,0,0)=1, \quad F(1)=F(0,0,1)=0, \quad F(2)=F(0,1,0)=1,\\
&F(3)=F(0,1,1)=0, \quad F(4)=F(1,0,0)=1, \quad F(5)=F(1,0,1)=0,\\
&F(6)=F(1,1,0)=0, \quad F(7)=F(1,1,1)=1.
\end{aligned}
$$

因此, F 的自然数形式为 $\mathbf{F}=(1,0,1,0,1,0,0,1)$.

于是 F 的多项式形式为

$$
\begin{aligned}
F(X)=&X^{B(0)}+X^{B(2)}+X^{B(4)}+X^{B(7)}X_1^0X_2^0X_3^0+X_1^0X_2^1X_3^0\\
&+X_1^1X_2^0X_3^0+X_1^1X_2^1X_3^1\\
=&(1+X_1)(1+X_2)(1+X_3)+(1+X_1)X_2(1+X_3)\\
&+X_1(1+X_2)(1+X_3)+X_1X_2X_3\\
=&1+X_3+X_1X_2.
\end{aligned}
$$

由于 $\mathbf{F}=\mathrm{IV}(r_F)$, 从向量表达式到多项式表达式的转换, 可以利用从自然数表达式到多项式表达式的转换公式. 下面给出一个更简单直接的公式. 它本质上是析取范式. 如果把析取范式写成矩阵形式, 则有

$$
\begin{aligned}
F(X)&=r_F\begin{bmatrix}X_1\\ \neg X_1\end{bmatrix}\begin{bmatrix}X_2\\ \neg X_2\end{bmatrix}\cdots\begin{bmatrix}X_n\\ \neg X_n\end{bmatrix}\\
&=r_F\begin{bmatrix}X_1\\ 1+X_1\end{bmatrix}\begin{bmatrix}X_2\\ 1+X_2\end{bmatrix}\cdots\begin{bmatrix}X_n\\ 1+X_n\end{bmatrix}.
\end{aligned}
\tag{1.1.22}
$$

(1.1.22) 给出了一个简单的计算公式. 以下是一个数值例子.

例 1.1.5　回忆例 1.1.4,

$$
r_F=\mathrm{IV}(\mathbf{F})=(1,0,0,1,0,1,0,1).
$$

则

$$
F(X)=r_F\begin{bmatrix}X_1\\ 1+X_1\end{bmatrix}\begin{bmatrix}X_2\\ 1+X_2\end{bmatrix}\begin{bmatrix}X_3\\ 1+X_3\end{bmatrix}=X_1X_2+X_3+1.
$$

这与例 1.1.4 中得到的结果一样.

最后, 我们讨论如何从多项式形式回到向量形式.

首先, F 可表示为

$$
\begin{aligned}
F(X)=M_Fx&=r_F\begin{bmatrix}X_1\\ X_1+1\end{bmatrix}\begin{bmatrix}X_2\\ X_2+1\end{bmatrix}\cdots\begin{bmatrix}X_n\\ X_n+1\end{bmatrix}\\
&=r_F\begin{bmatrix}0&1\\1&1\end{bmatrix}\begin{bmatrix}1\\ X_1\end{bmatrix}\begin{bmatrix}0&1\\1&1\end{bmatrix}\begin{bmatrix}1\\ X_2\end{bmatrix}\cdots\begin{bmatrix}0&1\\1&1\end{bmatrix}\begin{bmatrix}1\\ X_n\end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
&= r_F \left(\underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}}_{n} \right) \begin{bmatrix} 1 \\ X_1 \end{bmatrix} \begin{bmatrix} 1 \\ X_2 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ X_n \end{bmatrix} \\
&:= r_F P_n \xi_n := \alpha \xi_n,
\end{aligned} \tag{1.1.23}
$$

这里,

$$
\alpha = r_F P_n, \tag{1.1.24}
$$

$$
P_n = \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}}_{n}, \tag{1.1.25}
$$

$$
\xi_n = \begin{bmatrix} 1 \\ X_1 \end{bmatrix} \begin{bmatrix} 1 \\ X_2 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ X_n \end{bmatrix}. \tag{1.1.26}
$$

不难看出, ξ_n 是 $\mathbb{Z}_2^n$ 上的多项式的一个基底.

下面考虑 n 次多项式的一个自然基底, 它按自然顺序与增幂形式排列, 即

$$
\eta_n = \begin{bmatrix} 1 \\ X_1 \\ \vdots \\ X_n \\ X_1X_2 \\ \vdots \\ X_{n-1}X_n \\ X_1X_2X_3 \\ \vdots \\ X_{n-2}X_{n-1}X_n \\ \vdots \\ X_1X_2\cdots X_n \end{bmatrix}. \tag{1.1.27}
$$

那么, 任何 $F \in P(\mathbb{Z}_2^n)$ 可表示成

$$
F(X) = \beta \eta_n. \tag{1.1.28}
$$

下面讨论这两个基底 ξ_n 与 η_n 的关系.

容易看出, 这两个基底的元素是一样的, 只是排列顺序不同. 记 $\mu_{i_1,i_2,\cdots,i_r}$, $1 \leqslant i_1 < i_2 < \cdots < i_r \leqslant n$ 为 $X_{i_1}X_{i_2}\cdots X_{i_r}$ 在 ξ_n 中的位置. 为发现规律, 我们逐步计算.

第一步:

$$\begin{bmatrix} 1 \\ X_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ X_n \end{bmatrix} = \begin{bmatrix} 1 \\ X_n \\ X_{n-1} \\ X_{n-1}X_n \end{bmatrix}. \tag{1.1.29}$$

可得 $\mu_n = 2^0 + 1$, $\mu_{n-1} = 2^1 + 1$, $\mu_{n-1,n} = 2^1 + 2^0 + 1$.

第二步:

$$\begin{bmatrix} 1 \\ X_{n-2} \end{bmatrix} \begin{bmatrix} 1 \\ X_{n-1} \end{bmatrix} \begin{bmatrix} 1 \\ X_n \end{bmatrix} = \begin{bmatrix} 1 \\ X_{n-2} \end{bmatrix} \begin{bmatrix} 1 \\ X_n \\ X_{n-1} \\ X_{n-1}X_n \end{bmatrix} = \begin{bmatrix} 1 \\ X_n \\ X_{n-1} \\ X_{n-1}X_n \\ X_{n-2} \\ X_{n-2}X_n \\ X_{n-2}X_{n-1} \\ X_{n-2}X_{n-1}X_n \end{bmatrix}. \tag{1.1.30}$$

可得 $\mu_{n-2} = 2^2+1$, $\mu_{n-2,n} = 2^2+2^0+1$, $\mu_{n-2,n-1} = 2^2+2^1+1$, $\mu_{n-2,n-1,n} = 2^2 + 2^1 + 2^0 + 1$. 并且, 第一次得到的参数不变.

……

最后一步:

$$\begin{bmatrix} 1 \\ X_1 \end{bmatrix} \begin{bmatrix} 1 \\ X_2 \end{bmatrix} \cdots \begin{bmatrix} 1 \\ X_n \end{bmatrix} = \xi_n. \tag{1.1.31}$$

由以上观察不难看出如下结果.

命题 1.1.5

$$\mu_{i_1,i_2,\cdots,i_r} = \sum_{j=1}^{r} 2^{n-i_j} + 1. \tag{1.1.32}$$

证明 对于任何 $0 \leqslant j < n$, X_{n-j} 出现的位置是 $\mu_{n-j}=2^j + 1$. 换言之, X_{μ_1} 出现在 $2^{n-\mu_1} + 1$ 处. 也就是说, 它在 1 之后的 $2^{n-\mu_1}$ 处. 将 X_{μ_1} 看成 $(X_{\mu_1}, 1)$, 那么, (X_{μ_1}, X_{μ_2}) 应该出现在 X_{μ_1} 之后的 $2^{n-\mu_2}$ 处. 照此类推, 即得结论. □

根据命题 1.1.5, 我们可以构造一个矩阵 $\Psi_n \in \mathcal{L}_{2^n\times 2^n}$ 如下:

$$\Psi_n = \delta_{2^n}[1, \phi_1, \phi_2, \cdots, \phi_n], \tag{1.1.33}$$

这里

$$\begin{aligned} &\phi_1 = [\mu_1, \mu_2, \cdots, \mu_n],\\ &\phi_2 = [\mu_{1,2}, \mu_{1,3}, \cdots, \mu_{n-1,n}],\\ &\quad\cdots\cdots\\ &\phi_n = \mu_{1,2,\cdots,n}. \end{aligned}$$

利用这个 Ψ_n 可得

$$\Psi_n^{\mathrm{T}}\xi_n = \eta_n, \quad \eta_n = \Psi_n\xi_n. \tag{1.1.34}$$

对于函数 $F \in P(\mathbb{Z}_2^n)$, 记

$$F = \alpha\xi_n = \beta\eta_n,$$

则有

$$\begin{cases} \alpha = r_F P_n,\\ \beta = \alpha\Psi_n = r_F P_n \Psi_n,\\ r_F = \beta\Psi_n^{\mathrm{T}} P_n^{(-1)}, \end{cases} \tag{1.1.35}$$

这里 $P_n^{(-1)}$ 是布尔逆, 可写成

$$P_n^{(-1)} = \underbrace{\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}0 & 1\\ 1 & 1\end{bmatrix}}_{n}.$$

利用公式 (1.1.32) 容易算出 Ψ_n. 下面给出几个低维的 Ψ_n, 以便应用.

$$\begin{aligned} &\Psi_2 = \delta_4[1, 3, 2, 4],\\ &\Psi_3 = \delta_8[1, 5, 3, 2, 7, 6, 4, 8],\\ &\Psi_4 = \delta_{16}[1, 9, 5, 3, 2, 13, 11, 10, 7, 6, 4, 15, 14, 12, 8, 16],\\ &\Psi_5 = \delta_{32}[1, 17, 9, 5, 3, 2, 25, 21, 19, 18, 13, 11, 10, 7, 6, 4,\\ &\qquad\quad 29, 27, 26, 23, 22, 20, 15, 14, 12, 8, 31, 30, 28, 24, 16, 32]. \end{aligned} \tag{1.1.36}$$

公式 (1.1.35) 可用于从多项式表达式转换为向量表达式. 下面给一个例子.

例 1.1.6 考虑一个布尔函数

$$F(X) = 1 + X_1X_2 + X_1X_3 + X_1X_2X_3.$$

易知, 它可表示为

$$F(X) = [1,0,0,0,1,1,0,1]\eta := \beta\eta.$$

利用公式 (1.1.35) 可得

$$r_F = \beta\Psi_3^{\mathrm{T}} P_3^{(-1)} = [0,0,0,1,1,1,1,1].$$

令 $x = \vec{X}$, 则

$$f(x) = \vec{F}(X) = M_F x = \delta_2[2,2,2,1,1,1,1,1]x.$$

1.1.3 Walsh 变换

定义 1.1.4 设 $X = \{X_1, \cdots, X_n\}, \omega = \{\omega_1, \cdots, \omega_n\} \in \mathbb{Z}_2^n$.

(i) X 与 ω 的内积定义为

$$\langle x, \omega\rangle = x_1\omega_1 + \cdots + x_n\omega_n \in \mathbb{Z}_2. \tag{1.1.37}$$

(ii) 给定 $\omega \in \mathbb{Z}_2^n$. 定义一个函数 $\mathbb{Z}_2^n \to \mathbb{Z}_2$ 如下:

$$Q_\omega(X) = (-1)^{\langle\omega, X\rangle}, \quad X \in \mathbb{Z}_2^n. \tag{1.1.38}$$

写成序列形式, 则有

$$\mathbf{Q}_\omega = \left(Q_\omega(B(0)), Q_\omega(B(1)), \cdots, Q_\omega(B(2^n-1))\right),$$

这里, $B(X) \in \mathcal{B}^n$ 是 X 的二进制表达式.

引理 1.1.1 设 $\omega \neq 0$, 那么,

$$\sum_{i=0}^{2^n-1} (-1)^{\langle\omega, B(i)\rangle} = 0. \tag{1.1.39}$$

证明 设 $\omega_i \neq 0$. 对于每个 $B(i) = (X_1, \cdots, X_n)$ 满足 $\langle\omega, B(i)\rangle = 0$, 我们总可以构造一个 i^*, 使得 $B(i^*) = (X_1, \cdots, \neg X_i, \cdots, X_n)$. 于是 $\langle\omega, B(i^*)\rangle = 1$. 因为 $i \leftrightarrow i^*$ 是一一对应的, 则

$$\begin{aligned} &\left|\{i \,|\, 0 \leqslant i \leqslant 2^n-1 \text{ 且 } \langle\omega, B(i)\rangle = 0\}\right| \\ =& \left|\{i \,|\, 0 \leqslant i \leqslant 2^n-1 \text{ 且 } \langle\omega, B(i)\rangle = 1\}\right|, \end{aligned}$$

立得 (1.1.39). □

命题 1.1.6 $\{\mathbf{Q}_\omega := \mathbf{Q}_{B(\omega)} \,|\, \omega = 0, 1, \cdots, 2^n-1\}$ 是一组正交函数, 即

$$\langle\mathbf{Q}_\alpha, \mathbf{Q}_\beta\rangle = \begin{cases} 2^n, & \alpha = \beta, \\ 0, & \alpha \neq \beta. \end{cases} \tag{1.1.40}$$

证明 (i) 情况一: $\alpha = \beta$. 由条件知

$$\begin{aligned}\langle \mathbf{Q}_\alpha, \mathbf{Q}_\beta \rangle &= \sum_{i=0}^{2^n-1} (-1)^{\langle B(\alpha), B(i) \rangle} (-1)^{\langle B(\beta), B(i) \rangle} \\ &= \sum_{i=0}^{2^n-1} (-1)^{2\langle B(\alpha), B(i) \rangle} \\ &= \sum_{i=0}^{2^n-1} (-1)^0 = 2^n.\end{aligned}$$

(ii) 情况二: $\alpha \neq \beta$. 由于 $\alpha \neq \beta$, $\alpha + \beta \neq 0$. 利用引理 1.1.1 可得

$$\langle \mathbf{Q}_\alpha, \mathbf{Q}_\beta \rangle = \sum_{i=0}^{2^n-1} (-1)^{\langle B(\alpha+\beta), B(i) \rangle} = 0.$$ □

由于 $\{\mathbf{Q}_\omega \,|\, \omega = 0, 1, \cdots, 2^n - 1\}$ 是一族正交函数, 并由个数显见其完备, 于是对任一 $F \in P(\mathbb{Z}_2^n)$ 均有如下的分解:

$$\mathbf{F} = \sum_{\omega=0}^{2^n-1} S_F(\omega) \mathbf{Q}_\omega. \tag{1.1.41}$$

定义 1.1.5

$$\{S_F(\omega) \mid \omega = 0, 1, \cdots, 2^n - 1\}$$

称为 F 的第一 Walsh 变换.

命题 1.1.7 第一 Walsh 变换可计算如下:

$$S_F(\omega) = 2^{-n} \sum_{X=0}^{2^n-1} F(X) Q_X(\omega). \tag{1.1.42}$$

证明 任给 $\omega_0 \in \mathbb{Z}_2^n$, 利用 (1.1.41),

$$\begin{aligned}\langle \mathbf{F}, \mathbf{Q}_{\omega_0} \rangle &= \left\langle \sum_{\omega=0}^{2^n-1} S_F(\omega) \mathbf{Q}_\omega, \mathbf{Q}_{\omega_0} \right\rangle \\ &= S_F(\omega_0) \langle \mathbf{Q}_{\omega_0}, \mathbf{Q}_{\omega_0} \rangle \\ &= 2^n S_F(\omega_0).\end{aligned} \tag{1.1.43}$$

另一方面, 直接计算可得

$$\langle \mathbf{F}, \mathbf{Q}_{\omega_0} \rangle = \sum_{X=0}^{2^n-1} Q_{\omega_0}(X) F(X).$$

因此,

$$S_F(\omega_0) = 2^{-n}\sum_{X=0}^{2^n-1} Q_{\omega_0}(X)F(X). \qquad \square$$

下面介绍第二 Walsh 变换. 定义

$$G(X) := 1 - 2F(X). \tag{1.1.44}$$

我们将 $\bar{G}$ 在基底

$$\{\mathbf{Q}_\omega \mid \omega = 0, 1, \cdots, 2^n - 1\}$$

上展开,

$$\bar{G} = \sum_{\omega=0}^{2^n-1} S_{(F)}(\omega)\mathbf{Q}_\omega. \tag{1.1.45}$$

类似对第一 Walsh 变换的讨论, 对给定 $\omega_0 \in \mathbb{Z}_2^n$, 我们有

$$\langle \bar{G}, \mathbf{Q}_{\omega_0} \rangle = \sum_{X=0}^{2^n-1} (1 - 2F(X))Q_{\omega_0}(X). \tag{1.1.46}$$

由于 $F(X) \in \{0, 1\}$, 可知

$$(-1)^{F(X)} = 1 - 2F(X). \tag{1.1.47}$$

因此,

$$\langle \bar{G}, \mathbf{Q}_{\omega_0} \rangle = \sum_{X=0}^{2^n-1} (-1)^{F(X)} Q_{\omega_0}(X). \tag{1.1.48}$$

另一方面, 利用 (1.1.45) 可得

$$\langle \bar{G}, \mathbf{Q}_{\omega_0} \rangle = \left\langle \sum_{\omega=0}^{2^n-1} S_{(F)}(\omega)\mathbf{Q}_\omega, Q_{\omega_0} \right\rangle = 2^n \times S_{(F)}(\omega_0).$$

由于 ω_0 是任意的, 于是有

$$S_{(F)}(\omega) = \frac{1}{2^n}\sum_{X=0}^{2^n-1} (-1)^{F(X)} Q_\omega(X).$$

定义 1.1.6 设 $F(X) \in P(\mathbb{Z}_2^n)$, 则

$$S_{(F)}(\omega) = \frac{1}{2^n}\sum_{X=0}^{2^n-1} (-1)^{F(X)} Q_\omega(X) \tag{1.1.49}$$

称为 f 的第二 Walsh 变换.

利用 (1.1.44) 和 (1.1.45) 可得

$$F(X)=\frac{1}{2}-\frac{1}{2}\sum_{\omega=0}^{2^n-1}S_{(F)}(\omega)Q_\omega(X). \tag{1.1.50}$$

下面考虑两类 Walsh 变换的关系.

命题 1.1.8　$S_{(F)}(\omega)$ 和 $S_F(\omega)$ 有如下关系:

$$S_{(F)}(\omega)=\begin{cases}-2S_F(\omega), & \omega\neq 0,\\ 1-2S_F(\omega), & \omega=0.\end{cases} \tag{1.1.51}$$

证明　利用 (1.1.47) 可得

$$\begin{aligned}S_{(F)}(\omega)&=\frac{1}{2^n}\sum_{X=0}^{2^n-1}(-1)^{f(X)}Q_\omega(X)\\&=\frac{1}{2^n}\sum_{X=0}^{2^n-1}(1-2F(X))Q_\omega(X)\\&=\frac{1}{2^n}\sum_{X=0}^{2^n-1}Q_\omega(X)-\frac{2}{2^n}\sum_{X=0}^{2^n-1}F(X)Q_\omega(X).\end{aligned} \tag{1.1.52}$$

根据引理 1.1.1, 即得 (1.1.51). □

下面讨论 Walsh 变换的一些重要性质.

命题 1.1.9　设 $S_F(\omega)$ 为 $F(X)$ 的 Walsh 变换, 则对任何 $a\in\mathbb{Z}_2^n$, $F(X+a)$ 的 Walsh 变换为

$$S_{F(X+a)}(\omega)=Q_a(\omega)S_F(\omega). \tag{1.1.53}$$

证明　依定义, $F(X+a)$ 的 Walsh 变换为

$$\begin{aligned}S_{F(X+a)}(\omega)&=2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega B(X)\rangle}F(X+a)\\&=(-1)^{2\langle\omega,a\rangle}\cdot 2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega,B(X)\rangle}F(X+a)\\&=(-1)^{\langle\omega,a\rangle}\cdot 2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega,B(X)+a\rangle}F(X+a)\\&=(-1)^{\langle\omega,a\rangle}\cdot 2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega,B(X)\rangle}F(X)\\&=Q_a(\omega)S_F(\omega).\end{aligned}$$

□

命题 1.1.10 设 $S_F(\omega)$ 为 $F(X)$ 的 Walsh 变换, $S_G(\omega)$ 为 $G(X)$ 的 Walsh 变换, 则 $aF(X)+bG(X)$ 的 Walsh 变换为

$$S_{aF+bG}(\omega)=aS_F(\omega)+bS_G(\omega),\quad a,b\in\{0,1\}. \tag{1.1.54}$$

证明 依据定义, $aF(X)+bG(X)$ 的 Walsh 变换为

$$\begin{aligned}S_{aF+bG}(\omega)&=2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega,B(X)\rangle}(aF(X)+bG(X))\\&=a\cdot 2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega,B(X)\rangle}F(X)+b\cdot 2^{-n}\sum_{X=0}^{2^n-1}(-1)^{\langle\omega,B(X)\rangle}G(X)\\&=aS_F(\omega)+bS_G(\omega).\end{aligned}$$

□

命题 1.1.11 (Plancheral 等式)

$$\sum_{\omega=0}^{2^n-1}S_F^2(\omega)=S_F(0). \tag{1.1.55}$$

证明 因为

$$S_F(\omega)=2^{-n}\sum_{X=0}^{2^n-1}Q_\omega(X)F(X),$$

可得

$$\begin{aligned}\sum_{\omega=0}^{2^n-1}S_F^2(\omega)&=2^{-2n}\sum_{\omega=0}^{2^n-1}\sum_{X=0}^{2^n-1}Q_\omega(X)F(X)S_F(\omega)\\&=2^{-2n}\sum_{X=0}^{2^n-1}F(X)\left(\sum_{\omega=0}^{2^n-1}Q_\omega(X)S_F(\omega)\right)\\&=2^{-2n}\sum_{X=0}^{2^n-1}F^2(X)\\&=2^{-2n}\sum_{X=0}^{2^n-1}F(X).\end{aligned}$$

另一方面,

$$S_F(0)=2^{-2n}\sum_{\omega=0}^{2^n-1}F(X)Q_X(0)=2^{-2n}\sum_{\omega=0}^{2^n-1}F(X).$$

因此,

$$\sum_{\omega=0}^{2^n-1}S_F^2(\omega)=S_F(0).$$

□

于是, 下面结果是显然的.

命题 1.1.12 (Parseval 等式)　第二 Walsh 变换满足以下关系:

$$\sum_{\omega=0}^{2^n-1} S_{(F)}^2(\omega) = 1. \tag{1.1.56}$$

下面建立 **F** 和它的 Walsh 变换 $\mathbf{S}_F$ 之间的矩阵变换式. 由于对称性, 我们记

$$Q(\omega, X) := Q_\omega(X).$$

根据 (1.1.41) 可得

$$\begin{aligned}\mathbf{F} &= (F(0), F(1), \cdots, F(2^n-1))\\ &= (S_F(0), S_F(1), \cdots, S_F(2^n-1))\\ &\quad\times\begin{bmatrix} Q(0,0) & Q(0,1) & \cdots & Q(0,2^n-1)\\ Q(1,0) & Q(1,1) & \cdots & Q(1,2^n-1)\\ \vdots & \vdots & & \vdots\\ Q(2^n-1,0) & Q(2^n-1,1) & \cdots & Q(2^n-1,2^n-1)\end{bmatrix}\\ &:= (S_f(0), S_f(1), \cdots, S_f(2^n-1))H_n.\end{aligned} \tag{1.1.57}$$

(1.1.57) 可简记为

$$\mathbf{F} = \mathbf{S}_F H_n. \tag{1.1.58}$$

定义 1.1.7　$A = (a_{ij}) \in \mathcal{M}_{s\times s}$ 称为一个 Hadamard 矩阵, 如果它满足

(i) $$a_{ij} = \pm 1, \quad 1 \leqslant i, j \leqslant s;$$

(ii) $$A^{\mathrm{T}}A = AA^{\mathrm{T}} = sI_s.$$

命题 1.1.13　由式 (1.1.57) 定义的转换矩阵 H_n 满足以下性质:

(i) $$H_{n+1} = H_1 \otimes H_n; \tag{1.1.59}$$

(ii) $$H_n H_n = 2^n I_{2^n}; \tag{1.1.60}$$

(iii) H_n 是一个 Hadamard 矩阵.

证明　(ii) 直接来自命题 1.1.6. 因此, (iii) 是显见的. 下面证明 (i). 考察 H_{n+1}. 它可以表示成

$$H_{n+1} = \begin{bmatrix} H_n^1 & H_n^2\\ H_n^2 & H_n^3\end{bmatrix},$$

这里

$$H_n^1 = \begin{bmatrix} Q(0,0) & \cdots & Q(0,2^n-1) \\ \vdots & & \vdots \\ Q(2^n-1,0) & \cdots & Q(2^n-1,2^n-1) \end{bmatrix};$$

$$H_n^2 = \begin{bmatrix} Q(0,2^n) & \cdots & Q(0,2^{n+1}-1) \\ \vdots & & \vdots \\ Q(2^n-1,2^n) & \cdots & Q(2^n-1,2^{n+1}-1) \end{bmatrix};$$

$$H_n^3 = \begin{bmatrix} Q(2^n,2^n) & \cdots & Q(2^n,2^{n+1}-1) \\ \vdots & & \vdots \\ Q(2^{n+1}-1,2^n) & \cdots & Q(2^{n+1}-1,2^{n+1}-1) \end{bmatrix}.$$

首先考察 H_n^1, 将它写成

$$H_n^1 = \left(Q^1(\omega, X)\right),$$

这里

$$X = (0, X_1, X_2, \cdots, X_{2^n}), \quad \omega = (0, \omega_1, \omega_2, \cdots, \omega_{2^n}),$$

并且, 让 $(X_1, X_2, \cdots, X_{2^n})$ 和 $(\omega_1, \omega_2, \cdots, \omega_{2^n})$ 从 $(0, \cdots, 0)$ 变到 $(1, \cdots, 1)$. 则可验证 $H_n^1 = H_n$.

其次, 考虑 H_n^2, 将它记为

$$H_n^2 = \left(Q^2(\omega, X)\right),$$

这里

$$X = (1, X_1, X_2, \cdots, X_{2^n}), \quad \omega = (0, \omega_1, \omega_2, \cdots, \omega_{2^n}),$$

并且, 令 $(X_1, X_2, \cdots, X_{2^n})$ 及 $(\omega_1, \omega_2, \cdots, \omega_{2^n})$ 从 $(0, \cdots, 0)$ 变到 $(1, \cdots, 1)$. 则可见 $H_n^2 = H_n$.

最后, 我们考虑 H_n^3. 记其为

$$H_n^3 = \left(Q^3(\omega, X)\right),$$

这里,

$$X = (1, X_1, X_2, \cdots, X_{2^n}), \quad \omega = (1, \omega_1, \omega_2, \cdots, \omega_{2^n}).$$

类似的讨论可得 $H_n^3 = -H_n$.

注意到

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

于是

$$H_{n+1} = H_1 \otimes H_n.$$

□

下面给出一个例子.

例 1.1.7　考察逻辑函数

$$F(X) = X_1 \wedge \neg(X_2 \to X_3).$$

则

$$\mathbf{F} = (F(0), F(1), \cdots, F(7)) = (0,0,0,0,0,0,1,0).$$

利用 (1.1.58), 有

$$\mathbf{F} = \mathbf{S}_F H_3.$$

根据命题 1.1.13, 我们有 $H_3 H_3 = 8I_3$, 因此,

$$\mathbf{S}_F = \frac{1}{8}\mathbf{F} H_3.$$

由 (1.1.59),

$$H_3 = H_1 \otimes H_1 \otimes H_1 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}.$$

因此,

$$\mathbf{S}_F = \left(\frac{1}{8}, \frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, -\frac{1}{8}, \frac{1}{8}, \frac{1}{8}\right).$$

根据命题 1.1.8,

$$\mathbf{S}_{(F)} = \left(\frac{3}{4}, -\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}\right).$$

再者, 不难验证 Plancheral 等式 (1.1.55) 和 Parseval 等式 (1.1.56) 均成立.

1.2　布尔函数的线性性与对称性

1.2.1　线性性

定义 1.2.1　令 $F:\mathbb{Z}_2^n\to\mathbb{Z}_2$ 为一给定布尔函数.

(i)
$$F(X)=a_0+a_1X_1+a_2X_2+\cdots+a_nX_n,\quad a_i\in\mathbb{Z}_2$$
称为线性布尔函数. n 元线性布尔函数集合记作 $\mathcal{B}_n^{\mathcal{A}}$.

(ii) $a\in\mathbb{Z}_2^n$ 称为 F 的不变线性结构, 如果
$$F(X+a)+f(X)=0. \tag{1.2.1}$$

(iii) $a\in\mathbb{Z}_2^n$ 称为 F 的变线性结构, 如果
$$F(X+a)+F(X)=1. \tag{1.2.2}$$

(iv) 定义
$$\begin{aligned}&E_0:=\{a\in\mathbb{Z}_2^n\mid F(X+a)+F(a)=0\},\\&E_1:=\{a\in\mathbb{Z}_2^n\mid F(X+a)+F(a)=1\},\\&E:=E_0\cup E_1,\end{aligned} \tag{1.2.3}$$
那么, E 称为 F 的线性结构子空间.

命题 1.2.1　(i) $E_0\cap E_1=\varnothing$.

(ii) E 是一个向量空间, $E_0\subset E$ 是一个向量子空间.

(iii) 设 $E_1\neq\varnothing$, 则 $E_1=E_0+b$, 这里 $b\in E_1$.

证明　(i) 直接来自定义.

(ii) 分三种情况讨论.

(1) 设 $a,b\in E_0$, 那么,
$$\begin{aligned}&F(X+a+b)+F(X)\\&=F(X+a+b)+F(X+a)+F(X+a)+F(X)\\&=0+0=0,\end{aligned}$$
即 $a+b\in E_0\subset E$.

(2) 令 $a\in E_0$, $b\in E_1$, 那么,
$$\begin{aligned}&F(X+a+b)+F(X)\\&=F(X+a+b)+F(X+a)+F(X+a)+F(X)\\&=1+0=1.\end{aligned}$$

即 $a+b\in E_1\subset E$.

(3) 令 $a,b\in E_1$, 那么,

$$\begin{aligned}&F(X+a+b)+F(X)\\=&F(X+a+b)+F(X+a)+F(X+a)+F(X)\\=&1+1=0,\end{aligned}$$

即 $a+b\in E_0\subset E$.

因此, E 是一个向量空间. 并且, 情形 (i) 说明 E_0 是一个向量子空间.

(iii) 前面已经证明, 如果 $a\in E_0$, $b\in E_1$, 那么 $a+b\in E_1$. 这表明, $E_0+b\subset E_1$. 现在设 $\xi\in E_1$, 那么, $\xi+b\in E_0$, 并且

$$\xi=(\xi+b)+b\in E_0+b,$$

这说明 $E_1\subset E_0+b$. 结论显见. □

由此可得如下推论.

推论 1.2.1 (i)

$$|E_0|=2^r, \tag{1.2.4}$$

这里 r 是 E_0 的维数.

(ii) 或者 $E_1=\varnothing$, 或者

$$|E_1|=|E_0|. \tag{1.2.5}$$

证明 (i) 作为一个子空间, (1.2.4) 显然.

(ii) 设 $E_1\neq\varnothing$, 并令 $b\in E_1$. 定义 $\pi_b:E_0\to E_1$ 如下: $X\mapsto X+b$, 那么, 容易证明 π 是一对一且映上的. □

定义 1.2.2 给定一个布尔函数 F.

(i) 如果 $|E|=2^q$, 那么, q 称为 F 的线性结构的维数. 当 $q>0$ 时, 称 F 为一个具有线性结构的逻辑函数 (LFLS).

(ii) 设 F 为具有线性结构的逻辑函数. 如果 $E_0\neq\{0\}$, 那么 F 称为 I 型具有线性结构的逻辑函数. 如果 $E_0=\{0\}$, 那么 F 称为 II 型具有线性结构的逻辑函数.

下面讨论如何计算 E_0 和 E_1. 设 $F:\mathbb{Z}_2^n\to\mathbb{Z}_2$ 为一逻辑函数, 其结构矩阵为 $M_F\in\mathcal{L}_{2\times 2^n}$. 记 $\alpha=\ltimes_{i=1}^n a_i$, $X=\ltimes_{i=1}^n X_i$, 则显然 $(a_1,\cdots,a_n)\in E_1$, 当且仅当,

$$M_FM_pa_1X_1M_pa_2X_2\cdots M_pa_nX_n=M_FX_1X_2\cdots X_n, \tag{1.2.6}$$

这里 M_p 是 $+$ 的结构矩阵.

直接计算可知 (1.2.6) 等价于

$$M_F M_p \ltimes_{i=1}^{n-1} (I_{2^{2i}} \otimes M_p) \ltimes_{i=1}^{n-1} \left(I_{2^i} \otimes W_{[2,2^i]}\right) \alpha X = M_F X. \tag{1.2.7}$$

定义

$$\Theta_F := M_F M_p \ltimes_{i=1}^{n-1} (I_{2^{2i}} \otimes M_p) \ltimes_{i=1}^{n-1} \left(I_{2^i} \otimes W_{[2,2^i]}\right),$$

再将其等分成 2^n 块如下:

$$\Theta_F = [\psi_1, \psi_2, \cdots, \psi_{2^n}],$$

这里 $\psi_k = \mathrm{Blk}_k(\Theta_F)$, $k = 1, 2, \cdots, 2^n$. 那么, 简单计算就能验证以下结果.

定理 1.2.1 设 $\alpha = \ltimes_{i=1}^{n} a_i := \delta_{2^n}^{i}$. 那么 $(a_1, \cdots, a_n) \in E_0$, 当且仅当, $\psi_i = M_F$. $(a_1, \cdots, a_n) \in E_1$, 当且仅当, $\psi_i = M_n M_F$.

例 1.2.1 (i) 设 F 的结构矩阵为

$$M_F = \delta_2[2, 2, 2, 1, 2, 1, 1, 1].$$

那么

$$M_n M_F = \delta_2[1, 1, 1, 2, 1, 2, 2, 2].$$

不难算得

$$\begin{aligned}\Theta_F = \delta_2[&1, 1, 1, 2, 1, 2, 2, 2, 1, 1, 2, 1, 2, 1, 2, 2,\\ &1, 2, 1, 1, 2, 2, 1, 2, 2, 1, 1, 1, 2, 2, 2, 1,\\ &1, 2, 2, 2, 1, 1, 1, 2, 2, 1, 2, 2, 1, 1, 2, 1,\\ &2, 2, 1, 2, 1, 2, 1, 1, 2, 2, 2, 1, 2, 1, 1, 1].\end{aligned}$$

由于 $\psi_8 = M_F$ 且 $\psi_1 = M_n M_F$, 我们有

$$(0, 0, 0, 0, 0, 0, 0, 0) \in E_0,$$

且

$$(1, 1, 1, 1, 1, 1, 1, 1) \in E_1.$$

现在 $|E_0| = |E_1| = 1$, 因此, $|E| = 2$, $\dim(E) = 1$, 最后可得 F 具有线性结构的逻辑函数.

(ii) 设 F 的结构矩阵为

$$M_F = \delta_2[2, 2, 2, 2, 2, 2, 2, 1].$$

那么

$$M_n M_F = \delta_2[1, 1, 1, 1, 1, 1, 1, 2].$$

不难算得

$$\Theta_F = \delta_2[1,2,2,2,2,2,2,2,2,1,2,2,2,2,2,2,\\ 2,2,1,2,2,2,2,2,2,2,2,1,2,2,2,2,\\ 2,2,2,2,1,2,2,2,2,2,2,2,2,1,2,2,\\ 2,2,2,2,2,2,1,2,2,2,2,2,2,2,2,1].$$

因为只有 $\psi_8 = M_f$, 我们有

$$E_0 = \{(0,0,0,0,0,0,0,0)\}$$

且

$$E_1 = \varnothing.$$

现在 $|E_0| = 1$ 且 $|E_1| = 0$, 因此, $|E| = 1$. 最后可得 F 不是具有线性结构的逻辑函数.

1.2.2　非线性性

定义 1.2.3　(i) 设 $X = (X_1, \cdots, X_n) \in \mathcal{B}^n$. X 的汉明 (Hamming) 权重定义为[17]

$$w_H(X) = \big|\{i \mid X_i \neq 0\}\big|. \tag{1.2.8}$$

(ii) 设 $F, G \in \mathcal{B}_n^{\mathcal{F}}$, 则 F 与 G 的汉明距离定义为

$$d_H(F,G) := w_H(\mathbf{F} + \mathbf{F}). \tag{1.2.9}$$

利用汉明距离可定义非线性度.

定义 1.2.4　设 $F \in \mathcal{B}^{\mathcal{F}^n}$.

(i) F 的非线性度, 记作 N_F, 定义为

$$N_F := \min_{L \in \mathcal{B}_n^{\mathcal{A}}} d_H(\mathbf{F}, \bar{L}). \tag{1.2.10}$$

(ii) F 的线性度, 记作 C_F, 定义为

$$C_F := \max_{L \in \mathcal{B}_n^{\mathcal{A}}} d_H(\mathbf{F}, \bar{L}). \tag{1.2.11}$$

定义 1.2.5　设 $L(X) \in \mathcal{B}_n^{\mathcal{A}}$ 满足

$$d_H(\bar{L}, \mathbf{F}) = N_F, \tag{1.2.12}$$

则 $L(X)$ 称为 $F(X)$ 的最佳线性逼近.

为计算 $F \in \mathcal{B}_n^{\mathcal{F}}$ 的非线性度, 考察 F 的线性等价性.

以下的定理通过 Walsh 变换计算布尔函数线性等价的概率.

定理 1.2.2 设 $\omega=(\omega_1,\omega_2,\cdots,\omega_n)$, $X=(X_1,X_2,\cdots,X_n)\in\mathbb{Z}_2^n$, $X\in\mathbb{Z}_2^n$ 均匀分布, 则有

$$P\left(\{X|F(X)=\langle\omega,X\rangle\}\right)=\frac{1+S_{(F)}(\omega)}{2}, \tag{1.2.13}$$

$$P\left(\{X|F(X)=\neg\langle\omega,X\rangle\}\right)=\frac{1-S_{(F)}(\omega)}{2}. \tag{1.2.14}$$

证明 因为

$$\begin{aligned}
S_{(F)}(X)&=2^{-1}\sum_{x=0}^{2^n-1}(-1)^{F(X)}(-1)^{\langle\omega,X\rangle}\\
&=2^{-1}\left(\sum_{\{X|F(X)=\langle\omega,X\rangle\}}(-1)^{F(X)+\langle\omega,X\rangle}+\sum_{\{X|F(X)\neq\langle\omega,X\rangle\}}(-1)^{F(X)+\langle\omega,X\rangle}\right)\\
&=2^{-1}\left(\sum_{\{X|F(X)=\langle\omega,X\rangle\}}1+\sum_{\{X|F(X)\neq\langle\omega,X\rangle\}}(-1)\right)\\
&=P\left(\{X|F(X)=\langle\omega,X\rangle\}\right)-P\left(\{X|F(X)\neq\langle\omega,X\rangle\}\right),
\end{aligned}$$

并且, $P\left(\{X|F(X)=\langle\omega,X\rangle\}\right)+P\left(\{X|F(X)\neq\langle\omega,X\rangle\}\right)=1$, 所以即得 (1.2.13) 及 (1.2.14). □

定理 1.2.3 设 $F\in\mathcal{B}_n^{\mathcal{F}}$, 记

$$a=\max_{0\leqslant\omega\leqslant 2^n-1}\left|S_{(F)}(\omega)\right|,$$

则

$$N_F=2^n\left(\frac{1-a}{2}\right) \tag{1.2.15}$$

且

$$C_F=2^n\left(\frac{1+a}{2}\right). \tag{1.2.16}$$

证明 对任一线性函数 $L(X)=\langle\omega,X\rangle+\omega_0$, 我们有

$$\begin{aligned}
&P\left(\{X|F(X)=L(X)\}\right)=P\left(\{X|F(X)=\langle\omega,X\rangle+\omega_0\}\right)\\
=&\begin{cases}P\left(\{X|F(X)=\langle\omega,X\rangle\}\right), & \omega_0=0,\\ P\left(\{X|F(X)=\langle\omega,X\rangle+1\}\right)=P\left(\{X|F(X)\neq\langle\omega,X\rangle\}\right), & \omega_0=1.\end{cases}
\end{aligned}$$

因此,

$$P(\{X|F(X)=L(X)\})=\begin{cases}P(\{X|F(X)=\langle\omega,X\rangle\}), & L(X)=\langle\omega,X\rangle,\\ P(\{X|F(X)\neq\langle\omega,X\rangle\}), & L(X)=\langle\omega,X\rangle+1.\end{cases}$$

根据定理 1.2.2, 我们有

$$P(\{X|F(X)=L(X)\})=\begin{cases}\dfrac{1+S_{(F)}(\omega)}{2}, & L(X)=\langle\omega,X\rangle,\\ \dfrac{1-S_{(F)}(\omega)}{2}, & L(X)=\langle\omega,X\rangle+1.\end{cases}$$

于是有

$$\max_{L\in\mathcal{B}_n^{\mathcal{A}}} P(\{X|F(X)=L(X)\})=\frac{1+a}{2}, \tag{1.2.17}$$

以及

$$\min_{L\in\mathcal{B}_n^{\mathcal{A}}} P(\{X|F(X)=L(X)\})=\frac{1-a}{2}. \tag{1.2.18}$$

利用 (1.2.17) 可得

$$\begin{aligned}N_F &= \min_{L\in\mathcal{B}_n^{\mathcal{A}}} w_H(F+L)\\ &= \min_{L\in\mathcal{B}_n^{\mathcal{A}}} 2^n P(\{X|F(X)\neq L(X)\})\\ &= \min_{L\in\mathcal{B}_n^{\mathcal{A}}} 2^n(1-P(\{X|F(X)=L(X)\}))\\ &= 2^n\left(1-\max_{L\in\mathcal{B}_n^{\mathcal{A}}} P(\{X|F(X)=L(X)\})\right)\\ &= 2^n\left(\frac{1-a}{2}\right).\end{aligned}$$

利用 (1.2.18), 类似的讨论可得

$$C_F=2^n\left(\frac{1+a}{2}\right).$$

□

由上述定理直接可得

推论 1.2.2　对任何 $F\in\mathcal{B}_n^{\mathcal{F}}$,

$$N_F+C_F=2^n. \tag{1.2.19}$$

由定理 1.2.3 可见, 当 a 达到它的最小值时 N_F 达到最大值. 因为

$$\sum_{\omega=0}^{2^n-1} S_{(F)}^2 = 1,$$

当 $\left|S_{(F)}\right| = \text{const}$ 时, a 达到最小值. 这时我们有

$$\left|S_{(F)}\right| = 2^{-\frac{n}{2}}. \tag{1.2.20}$$

于是可得

$$N_F = 2^n\left(\frac{1-2^{-\frac{n}{2}}}{2}\right) = 2^{n-1} - 2^{\frac{n}{2}-1}. \tag{1.2.21}$$

因此, 我们有如下结果.

命题 1.2.2

$$N_F \leqslant 2^{n-1} - 2^{\frac{n}{2}-1}. \tag{1.2.22}$$

当 (1.2.21) 成立时, F 有最高非线性度. 这种布尔函数称为最佳函数, 或完全非线性函数, 它们在密码学中特别有用[17].

1.3 布尔函数的对称性

记 $\mathbf{H}_n < \mathbf{S}_n$ 为对称群 $\mathbf{S}_n$ 的一个子群.

定义 1.3.1 一个布尔函数 $F(X) \in \mathcal{B}_n^{\mathcal{F}}$ 称为关于 $\mathbf{H}_n < \mathbf{S}_n$ 对称, 如果

$$F(X_{\sigma(1)},\cdots,X_{\sigma(n)}) = F(X_1,\cdots,X_n), \quad \forall \sigma \in \mathbf{H}_n. \tag{1.3.1}$$

$F(X)$ 的 i 次齐次项记为 $\pi_i(F)$, 那么, 我们有

定理 1.3.1 F 关于 $\mathbf{H}_n$ 对称, 当且仅当, $\pi_i(F)$, $i = 0,1,\cdots,n$ 关于 $\mathbf{H}_n$ 对称.

证明 充分性显见, 我们证明必要性. 设 F 关于 $\mathbf{H}_n$ 对称. 我们用反证法. 设至少存在一个 i, 使得 $\pi_i(F)$ 关于 $\mathbf{H}_n$ 不对称. 将 $\pi_i(F)$ 表示为

$$\pi_i(F) = \sum_{1\leqslant j_1<j_2<\cdots<j_i\leqslant n} c_{j_1\cdots j_i} X_{j_1}\cdots X_{j_i}. \tag{1.3.2}$$

注意到, 不是所有的 $c_{j_1\cdots j_i} = 0$. 否则, $\pi_i(f) = 0$, 因此, 它是对称的. 对任何 $c_{j_1\cdots j_i} = 1$, 如果对所有的 $\sigma \in \mathcal{H}$, $c_{\sigma(j_1)\cdots\sigma(j_i)} = 1$, 则其为对称的. 所以我们假定存在 $\sigma \in \mathbf{H}_n$ 使得 $c_{\sigma(j_1)\cdots\sigma(j_i)} = 0$. 令 $X_0 = (X_1,\cdots,X_n)$ 由下式决定

$$X_j = \begin{cases} 1, & j \in \{j_1,\cdots,j_i\}, \\ 0, & \text{其他}. \end{cases}$$

则显见

$$\pi_i(F)(X_1, \cdots, X_n) = 1,$$
$$\pi_i(F)(X_{\sigma(1)}, \cdots, X_{\sigma(n)}) = 0.$$

注意到

$$\pi_k(X_0) = 0, \quad k > i.$$

因此,

$$F(X_1, \cdots, X_n) \neq F(X_{\sigma(1)}, \cdots, X_{\sigma(n)}).$$

这导致矛盾. □

以下的推论是明显的.

推论 1.3.1 f 关于 $\mathbf{S}_n$ 对称, 当且仅当, 对每个 $1 \leqslant i \leqslant n-1$, i 次多次多项式的每一项系数都相等. 准确地说,

$$c_{j_1 \cdots j_i}, \quad 1 \leqslant j_1 < j_2 < \cdots < j_i \leqslant n$$

同为 1 或同为 0.

关于 $\mathbf{S}_n$ 对称也称为完全对称.

推论 1.3.2 在 $\mathcal{B}_n^{\mathcal{F}}$ 中共有 2^{n+1} 个完全对称的布尔函数.

证明 根据推论 1.3.1, 如果 F 关于 $\mathbf{S}_n$ 对称, 则或者 $\pi_i(F) \equiv 0$ 或者 $\pi_i(F) \not\equiv 0$. 准确地说, 我们有: 或者 $\pi_i(F) = 0$ 或者 $\pi_i(F) = 1$. 于是, 结论显见. □

我们不妨将 2^{n+1} 完全对称函数记作 $F_0, F_1, \cdots, F_{2^{n+1}-1}$, 这里, F_i 由以下方法确定: 将 i 写成二进制形式

$$B(i) = (i_n, i_{n-1}, \cdots, i_0).$$

那么,

$$F_i(X) = \sum_{j=0}^{n} i_j P_j(X), \quad i = 0, 1, \cdots, 2^{n+1} - 1, \tag{1.3.3}$$

这里

$$P_j(X) = \sum_{1 \leqslant k_1 < \cdots < k_j \leqslant n} \prod_{s=1}^{j} X_{k_s}.$$

则可得以下命题.

命题 1.3.1 设 $\{F_i \,|\, i = 0, 1, \cdots, 2^{n+1} - 1\}$ 为完全对称逻辑函数, 其指标由以上方式决定. 那么, F_i 的汉明权重为

$$w_H(F_i) = \sum_{j=0}^{n} i_j \binom{n}{j}, \quad i = 0, 1, \cdots, 2^{n+1} - 1. \tag{1.3.4}$$

证明 因为

$$w_H(\bar{P}_j) = \binom{n}{j}, \quad j = 0, 1, \cdots, n,$$

则结论直接来自 (1.3.3). □

要考虑对 $\mathbf{H}_n$ 的对称性, 需要一些新的概念.

定义 1.3.2 设 G 为一群, S 为一非空集. G 在 S 上的映射 $G \times S \to S$ 称为一个群作用, 如果它满足

(i)

$$e(s) = s, \quad \forall\, s \in S,$$

这里 e 是 G 的单位元.

(ii)

$$g_1(g_2(s)) = (g_1 g_2)(s), \quad \forall\, s \in S.$$

定义 1.3.3 设 G 作用在 S 上, 记 $s \in S$.

(i) S 在 G 的作用下的轨道定义为

$$Gs := \{g(s) \,|\, g \in G\}. \tag{1.3.5}$$

(ii) 固定 $s \in S$, 定义 s 的稳定子群为

$$G_s := \{g \in G \,|\, g(s) = s\}. \tag{1.3.6}$$

不难看出, 不同轨道 $\{Gs \,|\, s \in S\}$ 形成 S 的一个分割, 因为对两个点 $s_1, s_2 \in S$, 或者 $Gs_1 = Gs_2$ 或者 $Gs_1 \cap Gs_2 = \varnothing$.

群轨道有如下性质[41].

命题 1.3.2 轨道 Gs 长度为

$$|Gs| = \frac{|G|}{|G_s|}. \tag{1.3.7}$$

下面的定理给出轨道的数目.

定理 1.3.2 (Burnside 引理) 设 G 作用在 S 上形成 m 条轨道, 那么

$$m|G| = \sum_{g \in G} |\mathrm{Fix}(g)|. \tag{1.3.8}$$

这里,

$$\mathrm{Fix}(g) = \{s \,|\, g(s) = s\}.$$

构造一个集合序列:

$$S_i = \big\{\ \{j_1, \cdots, j_i\} \subset \mathbb{Z} \mid 1 \leqslant j_t \leqslant n, t = 1, \cdots, i; j_p \neq j_q, p \neq q \big\}, \quad i = 1, \cdots, n-1.$$

设 $\mathbf{H}_n < \mathbf{S}_n$, 那么, $\mathbf{H}_n$ 在 S_i 的作用定义为

$$\sigma(\{j_1, \cdots, j_i\}) := \{\sigma(j_1), \cdots, \sigma(j_i)\}, \quad \sigma \in \mathbf{H}_n. \tag{1.3.9}$$

利用定理 1.3.1 以及推论 1.3.1, 不难证明以下结构.

定理 1.3.3　设 $\mathbf{H}_n$ 在 S_i 上的作用产生的轨道数为 m_i, $i = 1, \cdots, n-1$, 那么, 在 $\mathbf{H}_n$ 作用下对称的布尔函数的数目为

$$m = 2^{2 + \sum\limits_{i=1}^{n-1} m_i}. \tag{1.3.10}$$

设 $\mathbf{C}_n < \mathbf{S}_n$ 是由 $(1, 2, \cdots, n)$ 生成的子群, 即 $\mathbf{C}_n = \langle(1, 2, \cdots, n)\rangle$, 它称为循环子群. $F \in \mathcal{B}_n^{\mathcal{F}}$ 称为循环对称, 如果它是对循环子群对称的.

例 1.3.1　设 $n = 4$, $\mathbf{C}_4 = \langle(1, 2, 3, 4)\rangle$ 是由 $(1, 2, 3, 4)$ 生成的循环子群, 那么,

$$\begin{aligned} S_1 &= \{1, 2, 3, 4\}, \\ S_2 &= \big\{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}\big\}, \\ S_3 &= \big\{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}, \{2, 3, 4\}\big\}. \end{aligned}$$

不难验证, S_1 或 S_3 只有一个轨道. S_2 有两个轨道, 为

$$\begin{aligned} &\{1, 2\} \to \{2, 3\} \to \{3, 4\} \to \{4, 1\} \to \{1, 2\}, \\ &\{1, 3\} \to \{2, 4\} \to \{1, 3\}. \end{aligned}$$

由此可知, 所有对循环子群 $\mathbf{C}_4 = \langle(1, 2, 3, 4)\rangle$ 对称的 $F \in \mathcal{B}_4^{\mathcal{F}}$ 可表示如下:

$$\begin{aligned} F(X) = a_0 &+ a_1(X_1 + X_2 + X_3 + X_4) \\ &+ a_2(X_1X_2 + X_2X_3 + X_3X_4 + X_4X_1) \\ &+ a_3(X_1X_3 + X_2X_4) \\ &+ a_4(X_1X_2X_3 + X_1X_2X_4 + X_1X_3X_4 + X_2X_3X_4), \end{aligned} \tag{1.3.11}$$

这里 $a_i = 0$ 或 1, $i = 0, 1, 2, 3, 4$.

1.4　Canalizing 函数

Canalizing 函数也称强制布尔函数, 它在基因调控网络中有着重要作用[73,75,76]. 本节的内容主要来自文献 [84].

1.4.1 验证 Canalizing 函数

定义 1.4.1 [113] (i) 一个布尔函数 $F \in \mathcal{B}_n^{\mathcal{F}}$ 称为一个 Canalizing 函数, 如果存在一个 $i \in \{1, \cdots, n\}$ 和两个逻辑值 $u, v \in \{0,1\}$ 使得对任何 $(X_1, X_2, \cdots, X_n) \in \mathcal{D}^n$, 只要 $X_i = u$ 则 $F(X_1, \cdots, X_n) = v$.

(ii) 自变量 X_i 称为 Canalizing 变量, u 和 v 分别称为 Canalizing 变量值与 Canalizing 函数值. F 称为 (u, v) 型 Canalizing 函数.

(iii) 如果 Canalizing 函数值 $v = 1$, 则称 F 为正 Canalizing 函数; 如果 $v = 0$, 则称 F 为负 Canalizing 函数.

定义两族逻辑函数, $S^n = \{S_i^n \mid i = 1, 2, \cdots, n\}$, $\mathbf{S}^n = \{\mathbf{S}_i^n \mid i = 1, 2, \cdots, n\}$, 其中

$$
\begin{aligned}
S_1^n &= I_2 \otimes \mathbf{1}_{2^{n-1}}^{\mathrm{T}}, \\
\mathbf{S}_1^n &= M_n \otimes \mathbf{1}_{2^{n-1}}^{\mathrm{T}}, \\
S_2^n &= \mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-2}}^{\mathrm{T}}, \\
\mathbf{S}_2^n &= \mathbf{1}_2^{\mathrm{T}} \otimes M_n \otimes \mathbf{1}_{2^{n-2}}^{\mathrm{T}}, \\
&\cdots\cdots \\
S_n^n &= \mathbf{1}_{2^{n-1}}^{\mathrm{T}} \otimes I_2, \\
\mathbf{S}_n^n &= \mathbf{1}_{2^{n-1}}^{\mathrm{T}} \otimes M_n,
\end{aligned}
\tag{1.4.1}
$$

这里, M_n 是 $\neg$ 的结构矩阵.

下面给出判定 Canalizing 函数的方法.

命题 1.4.1 设 $F(X) \in \mathcal{B}_n^{\mathcal{F}}$, 其结构矩阵为 $M_F \in \mathcal{L}_{2\times 2^n}$,

(i) $F(X)$ 是一个 (1,1) 型 Canalizing 函数, 当且仅当, 存在 $i \in \{1, \cdots, n\}$, 使得

$$
M_F(S_i^n)^{\mathrm{T}} = \begin{bmatrix} 2^{n-1} & * \\ 0 & * \end{bmatrix} \in \mathcal{M}_{2\times 2}. \tag{1.4.2}
$$

(ii) $F(X)$ 是一个 (1,0) 型 Canalizing 函数, 当且仅当, 存在 $i \in \{1, \cdots, n\}$, 使得

$$
M_F(S_i^n)^{\mathrm{T}} = \begin{bmatrix} 0 & * \\ 2^{n-1} & * \end{bmatrix}. \tag{1.4.3}
$$

(iii) $F(X)$ 是一个 (0,1) 型 Canalizing 函数, 当且仅当, 存在 $i \in \{1, \cdots, n\}$, 使得

$$
M_F(S_i^n)^{\mathrm{T}} = \begin{bmatrix} * & 2^{n-1} \\ * & 0 \end{bmatrix} \in \mathcal{M}_{2\times 2}. \tag{1.4.4}
$$

(iv) $f(X)$ 是一个 (0,0) 型 Canalizing 函数, 当且仅当, 存在 $i \in \{1, \cdots, n\}$, 使得

$$M_F (S_i^n)^{\mathrm{T}} = \begin{bmatrix} * & 0 \\ * & 2^{n-1} \end{bmatrix} \in \mathcal{M}_{2\times 2}, \tag{1.4.5}$$

这里 $*$ 表示无关紧要的数.

证明　我们只证 (i), 其他情况类似.

(必要性) 记 $x_i = \vec{X}_i$, $i = 1, 2, \cdots, n$, $x = \ltimes_{i=1}^n x_i$. 设 $F(X)$ 为 $(1,1)$ 型 Canalizing 函数, 假定 X_i 是 Canalizing 变量. 根据定义, 当 $x_i = \delta_2^1$ 时

$$f = \vec{F} = M_F x = \delta_2^1.$$

注意到 $x_j = \begin{bmatrix} X_j \\ 1 - X_j \end{bmatrix}$, 这里 $X_j \in \mathcal{D}$, $j = 1, 2, \cdots, n$. 先假定 $i = 1$, 即 $X_1 = 1$.

$$x = \begin{bmatrix} X_1 X_2 \cdots X_{n-1} X_n \\ X_1 X_2 \cdots X_{n-1}(1 - X_n) \\ \vdots \\ X_1 (1 - X_2) \cdots (1 - X_{n-1})(1 - X_n) \\ (1 - X_1) X_2 \cdots X_{n-1} X_n \\ (1 - X_1) X_2 \cdots X_{n-1} (1 - X_n) \\ \vdots \\ (1 - X_1)(1 - X_2) \cdots (1 - X_{n-1})(1 - X_n) \end{bmatrix}$$

$$= \begin{bmatrix} X_2 \cdots X_{n-1} X_n \\ X_2 \cdots X_{n-1} (1 - X_n) \\ \vdots \\ (1 - X_2) \cdots (1 - X_{n-1})(1 - X_n) \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{1.4.6}$$

注意到 $x \in \Delta_{2^n}$ 且 $x_1 = \delta_2^1$, 1 只可能出现在前 2^{n-1} 中的某个位置. 由

$$M_F x = \delta_2^1$$

可知 $(M_F)_{1,t}=1,\ t\in\{1,2,\cdots,2^{n-1}\}$. 由 S_1^n 的构造即得

$$M_f(S_1^n)^{\mathrm{T}}=\begin{bmatrix}2^{n-1} & * \\ 0 & *\end{bmatrix}.$$

对于一般情况, 设 X_i 为 Canalizing 变量, 我们不难知道 $(M_F)_{1,t}=1,\ t\in\{j\mid(S_i^n)_{1,j}=1\}$ 并且

$$M_F(S_i^n)^{\mathrm{T}}=\begin{bmatrix}2^{n-1} & * \\ 0 & *\end{bmatrix}. \tag{1.4.7}$$

(充分性) 设存在 i 使得 (1.4.7) 成立, 因为 S_i^n 第一行只有 2^{n-1} 个 1, 于是

$$(M_F)_{1,i}=1,\quad i\in\{j\mid(S_i^n)_{1,j}=1\}.$$

选择 $x=x_1\ltimes\cdots\ltimes x_{i-1}\ltimes\delta_2^1\ltimes x_{i+1}\ltimes\cdots\ltimes x_n$, 则得

$$f=\vec{F}=M_Fx=\delta_2^1.$$

□

由上述定理易知如下推论.

推论 1.4.1 一个逻辑函数 F, 其代数状态空间表示为 $f(x)=M_Fx$, 这里 $M_F\in\mathcal{L}_{2\times2^n}$, $F(X)$ 是 Canalizing 函数, 其 Canalizing 变量为 X_i, 当且仅当, 2^{n-1} 是 $M_F(S_i^n)^{\mathrm{T}}$ 的一个元素.

注意, $S=\{S_i^n\}$ 及 $\mathbf{S}=\{\mathbf{S}_i^n\}$ 中的矩阵均为 Canalizing 函数的结构矩阵. 一个 Canalizing 函数可能有多于一个的 Canalizing 变量.

设 $M,\ N\in\mathcal{L}_{2\times s}$, 记

$$\begin{aligned}&\mathrm{Col}_i(M\wedge N)=M_c\,\mathrm{Col}_i(M)\,\mathrm{Col}_i(N),\\&\mathrm{Col}_i(M\vee N)=M_d\,\mathrm{Col}_i(M)\,\mathrm{Col}_i(N),\quad 1\leqslant i\leqslant s.\end{aligned} \tag{1.4.8}$$

命题 1.4.2 一个逻辑函数 $F(X)$, 其代数状态空间表示为 $f(x)=M_Fx$, 这里 $M_F\in\mathcal{L}_{2\times2^n}$. 那么,

(i) F 是 (1,1) 型 Canalizing 函数, 当且仅当, 存在 S_i^n, $1\leqslant i\leqslant n$, 使得

$$M_F\vee S_i^n=M_F. \tag{1.4.9}$$

(ii) F 是 (1,0) 型 Canalizing 函数, 当且仅当, 存在 $\mathbf{S}_i^n$, $1\leqslant i\leqslant n$, 使得

$$M_F\vee\mathbf{S}_i^n=M_F. \tag{1.4.10}$$

(iii) F 是 (0,1) 型 Canalizing 函数, 当且仅当, 存在 S_i^n, $1 \leqslant i \leqslant n$, 使得

$$M_F \wedge S_i^n = M_F. \tag{1.4.11}$$

(iv) F 是 (0,0) 型 Canalizing 函数, 当且仅当, 存在 $\mathbf{S}_i^n$, $1 \leqslant i \leqslant n$, 使得

$$M_F \wedge \mathbf{S}_i^n = M_F. \tag{1.4.12}$$

证明　我们只证明 (i), (ii)—(iv) 的证明类似.

根据命题 1.4.1 可知, 如果 F 是 (1,1) 型 Canalizing 函数且其 Canalizing 变量为 X_i, 那么,

$$(M_F)_{1,i} = 1, \quad i \in \{j \mid (S_i^n)_{1,j} = 1\}.$$

直接计算即得 (1.4.9). □

1.4.2　Canalizing 函数的个数

Canalizing 函数的个数是一个有趣的问题, 最先是文献 [73] 给出一个上限的估计 $(4n \times 2^{2^{n-1}})$. 其后, 文献 [70] 用概率的方法给出了准确数. 利用 Canalizing 函数的准确表示, 本小节给出一种较为简洁的计算方法.

引理 1.4.1 (容斥原理)　设 $A_1, \cdots, A_n$ 为 n 给定集合, $A = A_1 \cup \cdots \cup A_n$. 那么, A 的势数, 记作 $|A|$, 可计算如下:

$$\begin{aligned}|A| = &\sum_{i=1}^n |A_i| - \sum_{1 \leqslant i_1 < i_2 \leqslant n} |A_{i_1} \cap A_{i_2}| + \cdots \\ &+ (-1)^{k+1} \sum_{1 \leqslant i_1 < \cdots < i_k \leqslant n} |A_{i_1} \cap A_{i_2} \cap \cdots \cap A_{i_k}| + \cdots \\ &+ (-1)^{n+1} |A_1 \cap A_2 \cap \cdots \cap A_n|.\end{aligned} \tag{1.4.13}$$

下面考虑 Canalizing 函数个数, 我们先计算 (1,1) 型 Canalizing 函数. 记

(i) E^{11}: (1,1) 型 Canalizing 函数集合.

(ii) E_i^{11}, $i = 1, \cdots, n$: (1,1) 型 Canalizing 函数集合, 其 Canalizing 变量为 X_i.

显然

$$E^{11} = \bigcup_{i=1}^n E_i^{11}.$$

利用引理 1.4.1,

$$|E^{11}| = \sum_{i=1}^n |E_i^{11}| - \sum_{1 \leqslant i_1 < i_2 \leqslant n} |E_{i_1}^{11} \cap E_{i_2}^{11}| + \cdots$$

$$
\begin{aligned}
&+(-1)^{k+1}\sum_{1\leqslant i_1<\cdots<i_k\leqslant n}|E_{i_1}^{11}\cap E_{i_2}^{11}\cap\cdots\cap E_{i_k}^{11}|+\cdots\\
&+(-1)^{n+1}|E_1^{11}\cap\cdots\cap E_n^{11}|. \qquad (1.4.14)
\end{aligned}
$$

如果 $f(x)=\vec{F}(X)=M_F x$ 为 (1,1) 型 Canalizing 函数且其 Canalizing 变量为 X_i, 由命题 1.4.2 可知

$$
M_F\wedge S_i^n=S_i^n.
$$

换言之, $F(X)$ 为 (1,1) 型 Canalizing 函数且其 Canalizing 变量为 X_i, 当且仅当,

$$
\mathrm{Col}_j(S_i^n)=\delta_2^1\Rightarrow \mathrm{Col}_j(M_F)=\delta_2^1. \tag{1.4.15}
$$

注意到

$$
\left|\{j\mid \mathrm{Col}_j(S_i^n)=\delta_2^1\}\right|=2^{n-1},
$$

则 M_F 有 2^{n-1} 可自由选择的列, 故

$$
\left|E_i^{11}\right|=2^{n-1},\quad i=1,\cdots,n. \tag{1.4.16}
$$

如果 $f(x)=\vec{F}(X)=M_F x$ 为 (1,1) 型 Canalizing 函数且其 Canalizing 变量既为 X_i 又为 X_j, $i\neq j$, 由命题 1.4.2 可知

$$
M_F\wedge(S_i^n\vee S_j^n)=S_i^n\vee S_j^n.
$$

换言之, $F(X)$ 为 (1,1) 型 Canalizing 函数且其 Canalizing 变量为 X_i 及 X_j, 当且仅当,

$$
\mathrm{Col}_j(S_i^n\vee S_j^n)=\delta_2^1\Rightarrow \mathrm{Col}_j(M_F)=\delta_2^1. \tag{1.4.17}
$$

注意到

$$
\left|\{k\mid \mathrm{Col}_k(S_i^n\vee S_j^n)=\delta_2^1\}\right|=2^{n-1}+2^{n-2},
$$

则 M_F 有 2^{n-2} 可自由选择的列, 故

$$
\left|E_i^{11}\cap E_j^{11}\right|=2^{n-2},\quad 1\leqslant i<j\leqslant n. \tag{1.4.18}
$$

同理可知

$$
\left|E_{i_1}^{11}\cap E_{i_2}^{11}\cap\cdots\cap E_{i_k}^{11}\right|=2^{n-k},\quad 1\leqslant i_1<\cdots<i_k\leqslant n;\quad k=1,2,\cdots,n. \tag{1.4.19}
$$

利用引理 1.4.1 即得以下命题.

命题 1.4.3

$$\left|E^{11}\right| = \sum_{k=1}^{n}(-1)^{k+1}\binom{n}{k}2^{n-k}. \tag{1.4.20}$$

记

$$E^1 = E^{11} \cup E^{01}; \quad E^0 = E^{10} \cup E^{00}.$$

则 E^1, E^0 分别为正、负 Canalizing 函数集合. 它们的势可计算如下.

命题 1.4.4

$$|E^1| = |E^0| = \sum_{m=1}^{n}(-1)^{m+1}\binom{n}{m}2^m(2^{2^{n-m}}-1)+1. \tag{1.4.21}$$

证明 我们计算 $|E^1|$, $|E^0|$ 同理可得.

考虑 $S^n \cup \mathbf{S}^n$, 显然,

$$|S^n \cup \mathbf{S}^n| = 2n. \tag{1.4.22}$$

注意到所有 $S^n \cup \mathbf{S}^n$ 中的矩阵都是正 Canalizing 矩阵. 其中只有一个定常正 Canalizing 矩阵, 记作

$$M_c = \mathbf{1}_{2^n}^{\mathrm{T}} \otimes \delta_2^1.$$

下面考虑非定常的 Canalizing 矩阵.

考察 $M \in E^1 \backslash M_c$. 类似于前面的讨论可知:

(i) 如果 $M \in S \cup \overline{S}$, 则由 M 生成的非定常正 Canalizing 矩阵的个数为

$$2\binom{n}{1}(2^{2^{n-1}}-1).$$

这里, 系数 2 来自 S 与 $\mathbf{S}$ 的两种选择, -1 是减去定常矩阵.

(ii) 如果 $M = M_1 \vee M_2$, 这里 $M_1,\ M_2 \in S \cup \overline{S}$, 那么, 由 $M_1,\ M_2$ 生成的非定常正 Canalizing 矩阵的个数为

$$\frac{1}{2}2^2\binom{n}{1}\binom{n-1}{1}(2^{2^{n-2}}-1).$$

(iii) 如果 $M = M_1 \vee M_2 \vee \cdots \vee M_k$, 这里 $M_1, M_2, \cdots, M_k \in S \cup \overline{S}$, 那么, 矩阵集合 $\{M_1, M_2, \cdots, M_k\}$ 的选择有

$$P_k = 2^k\frac{\binom{n}{1}\binom{n-1}{1}\cdots\binom{n-k+1}{1}}{k!}.$$

于是, 由 M_1, M_2, $\cdots$, M_m 生成的非定常正 Canalizing 矩阵的个数为

$$P_k 2^k (2^{2^{n-k}} - 1).$$

利用引理 1.4.1, 并加上唯一定常矩阵, 于是有

$$\begin{aligned}|E^1| &= \sum_{k=1}^{n} (-1)^{k+1} P_k (2^{2^{n-k}} - 1) + 1 \\ &= \sum_{k=1}^{n} (-1)^{k+1} 2^k \frac{n(n-1)\cdots(n-k+1)}{k!} (2^{2^{n-k}} - 1) + 1 \\ &= \sum_{k=1}^{n} (-1)^{k+1} \binom{n}{k} 2^k (2^{2^{n-k}} - 1) + 1.\end{aligned}$$

□

为了得到一般公式, 我们需要一些准备工作.

引理 1.4.2 设 $F(X)$ 是以 X_i 为 Canalizing 变量的正 Canalizing 函数, 同时它也是以 X_j 为 Canalizing 变量的负 Canalizing 函数, 那么 $i = j$.

证明 设 $i \neq j$, X_i 的 Canalizing 变量值为 ξ_i, X_j 的 Canalizing 变量值为 ξ_j, 那么

$$F(X_1, \cdots, X_i = \xi_i, \cdots, X_j = \xi_j, \cdots, X_n)$$

既等于 δ_2^1 又等于 δ_2^2, 矛盾. □

利用引理 1.4.1、引理 1.4.2 以及命题 1.4.4 可得如下推论.

推论 1.4.2 考虑一个逻辑函数 $F(X) \in \mathcal{B}_n^{\mathcal{F}}$, 其代数状态空间表达式为 $f(x) = M_F x$, 这里 $M_F \in \mathcal{L}_{2\times 2^n}$ 为其结构矩阵. 那么, $F(X)$ 既是正 Canalizing 函数又是负 Canalizing 函数, 当且仅当,

$$M_F \in S \cup \overline{S}.$$

现在我们可以考虑 Canalizing 函数集的势数了. 记

$$E^0 = E^{10} \cup E^{00}, \quad E = E^1 \cup E^0.$$

因为根据引理 1.4.1,

$$|E| = |E^1| + |E^0| - |E^1 \cap E^0|.$$

根据推论 1.4.2,

$$|E^1 \cap E^0| = 2n. \tag{1.4.23}$$

利用命题 1.4.4 及等式 (1.4.23),

$$
\begin{aligned}
|E| &= |E^1| + |E^0| - |E^1 \cap E^0| \\
&= 2\left(\sum_{k=1}^{n}(-1)^{k+1}\binom{n}{k}2^k(2^{2^{n-k}}-1)+1\right)-2n \\
&= 2\sum_{k=1}^{n}(-1)^{k+1}\binom{n}{k}2^k 2^{2^{n-k}} + 2(1+(-1)^n-n).
\end{aligned}
$$

将以上结论写成命题.

命题 1.4.5 Canalizing 函数 $F \in \mathcal{B}_n^{\mathcal{F}}$ 的集合 E 的势为

$$
|E| = 2\sum_{k=1}^{n}(-1)^{k+1}\binom{n}{k}2^k 2^{2^{n-k}} + 2(1+(-1)^n-n). \tag{1.4.24}
$$

注 1.4.1 公式 (1.4.24) 与文献 [70] 中给出的公式一致.

以下的算法说明如何生成 Canalizing 函数.

算法 1.4.1 选 $Q = S$ 或 $Q = \mathbf{S}$, $\sigma = \vee$ 或 $\sigma = \wedge$.

第一步: 任选 m 个矩阵 $M_1, \cdots, M_m \in Q$.

第二步: 计算 $M = M_1 \sigma \cdots \sigma M_m$.

第三步: 对矩阵 M, 如果 $\sigma = \vee$, 保持 δ_2^1 列不变, 如果 $\sigma = \wedge$, 保持 δ_2^2 列不变, 而将其系列随意地改为 δ_2^1 或 δ_2^2.

命题 1.4.6 算法 1.4.1 得到的即为至少有 m 个 Canalizing 变量的 Canalizing 函数.

(i) 当 $Q = S$, $\sigma = \vee$ 时, 得到的是 (1,1) 型 Canalizing 函数.

(ii) 当 $Q = S$, $\sigma = \wedge$ 时, 得到的是 (1,0) 型 Canalizing 函数.

(iii) 当 $Q = \mathbf{S}$, $\sigma = \vee$ 时, 得到的是 (0,1) 型 Canalizing 函数.

(iv) 当 $Q = \mathbf{S}$, $\sigma = \wedge$ 时, 得到的是 (0,0) 型 Canalizing 函数.

下面给出一个例子.

例 1.4.1 构造一个 (0,1) 型 Canalizing 函数 $F(X) \in \mathcal{B}_4^{\mathcal{F}}$, 其 Canalizing 变量为 X_2 及 X_3.

第一步: 选 $\bar{S}_2^4$ 及 $\bar{S}_3^4$ 如下

$$
\begin{aligned}
\bar{S}_2^4 &= \delta_2[2,2,2,2,1,1,1,1,2,2,2,2,1,1,1,1], \\
\bar{S}_3^4 &= \delta_2[2,2,1,1,2,2,1,1,2,2,1,1,2,2,1,1];
\end{aligned}
$$

第二步: 计算 M,

$$
M = \overline{S_2^4} +_1 \overline{S_3^4} = \delta_2[2,2,1,1,1,1,1,1,2,2,1,1,1,1,1,1];
$$

第三步: 固定 δ_2^1 的列, 任选其他列, 即或为 δ_2^1 或为 δ_2^2, 则得

$$M' = \begin{bmatrix} * & * & 1 & 1 & 1 & 1 & 1 & 1 & * & * & 1 & 1 & 1 & 1 & 1 & 1 \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 & * & * & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

则 M' 是一个 $(0,1)$ 型 Canalizing 矩阵, 其 Canalizing 变量为 X_2 及 X_3.

1.4.3 级联 Canalizing 函数

本节讨论级联 Canalizing 函数, 它在基因调控网络的讨论中被广泛引用[68,76].

定义 1.4.2 设 $F(X) \in \mathcal{B}_n^{\mathcal{F}}$, $\sigma \in \mathbf{S}_n$.

(i) $F(X)$ 称为依顺序 σ 级联的, 带有 Canalizing 输入值 $a_1, \cdots, a_n$ 和 Canalizing 输出值 $b_1, \cdots, b_n$ 的 Canalizing 函数, 如果

$$F(X) = \begin{cases} b_1, & \text{如果 } X_{\sigma(1)} = a_1, \\ b_2, & \text{如果 } X_{\sigma(1)} \neq a_1, \text{ 且 } X_{\sigma(2)} = a_2, \\ b_3, & \text{如果 } X_{\sigma(1)} \neq a_1,\ X_{\sigma(2)} \neq a_2, \text{ 且 } X_{\sigma(3)} = a_3, \\ & \quad \cdots\cdots \\ b_n, & \text{如果 } X_{\sigma(1)} \neq a_1,\ \cdots,\ X_{\sigma(n-1)} \neq a_{n-1} \text{ 且 } X_{\sigma(n)} = a_n, \\ *, & \text{如果 } X_{\sigma(1)} \neq a_1,\ \cdots,\ X_{\sigma(n)} \neq a_n. \end{cases} \tag{1.4.25}$$

(ii) $F(X)$ 称为依自然序级联 Canalizing 函数, 如果 $\sigma(i) = i$, $i = 1, 2, \cdots, n$.

直接由结构矩阵的构造可以得到以下结论.

命题 1.4.7 设 $F \in \mathcal{B}_n^{\mathcal{F}}$ 为依自然序的级联 Canalizing 函数, 满足 (1.4.25), 并且

$$a_1 = a_2 = \cdots = a_n = 1.$$

设 F 的结构矩阵为 $M_F \in \mathcal{L}_{2\times 2^n}$, 那么

$$\mathrm{Col}_i(M_F) = \begin{cases} \mu_1, & 1 \leqslant i \leqslant 2^{n-1}, \\ \mu_2, & 2^{n-1}+1 \leqslant i \leqslant 2^{n-1}+2^{n-2}, \\ & \quad \cdots\cdots \\ \mu_n, & 2^{n-1}+2^{n-2}+\cdots+2+1 \leqslant i \leqslant 2^{n-1}+2^{n-2}+\cdots+2+1, \end{cases} \tag{1.4.26}$$

这里 $\mu_i = \vec{b}_i$, $i = 1, 2, \cdots, n$.

例 1.4.2 $F \in \mathcal{B}_4^{\mathcal{F}}$ 为依自然序的级联 Canalizing 函数, 满足 (1.4.25), 并且

$$a_1 = a_2 = a_3 = a_4 = 1, \quad b_1 = b_3 = 1, \quad b_2 = b_4 = 0.$$

那么, F 的结构矩阵为

$$M_F=\delta_2[1,1,1,1,1,1,1,1,2,2,2,2,1,1,2,*]. \tag{1.4.27}$$

下面回忆置换矩阵 (详见卷一).

设 $(k_1,k_2,\cdots,k_n)\in\mathbb{Z}^n$, $k_j\geqslant 2$, $j=1,2,\cdots,n$. $\sigma\in\mathbf{S}_n$. 构造一组向量

$$D_k=\left\{\delta_{k_{\sigma(1)}}^{i_{\sigma(1)}}\delta_{k_{\sigma(2)}}^{\sigma(2)}\cdots\delta_{k_{\sigma(n)}}^{i_{\sigma(n)}}\mid 1\leqslant i_{\sigma(j)}\leqslant k_{\sigma(j)}\right\}.$$

将 D_k 中的向量依 $\mathrm{Id}\left(i_{\sigma^{-1}(1)},\cdots,i_{\sigma^{-1}(n)};k_{\sigma^{-1}(1)},\cdots,k_{\sigma^{-1}(n)}\right)$ 排列, 记生成的矩阵为 W_σ, 称为 σ 置换矩阵.

例 1.4.3　设 $k_1=k_2=k_3=k_4=2$, $\sigma=(2,4,1,3)$, 则

$$D_4=\left\{\delta_k^i\delta_k^j\delta_k^p\delta_k^q\mid i,j,p,q=1,2\right\}.$$

注意到 j 列为 $\sigma^{-1}(4)$, i 列为 $\sigma^{-1}(3)$, p 列为 $\sigma^{-1}(2)$, q 列为 $\sigma^{-1}(1)$, 所以 W_σ 为 D_4 中元素按 $\mathrm{Id}(q,p,i,j;2,2,2,2)$ 排列, 即

$$\begin{aligned}W_\sigma=&\big[\delta_2^1\delta_2^1\delta_2^1\delta_2^1,\delta_2^1\delta_2^2\delta_2^1\delta_2^1,\delta_2^2\delta_2^1\delta_2^1\delta_2^1,\delta_2^2\delta_2^2\delta_2^1\delta_2^1,\\&\delta_2^1\delta_2^1\delta_2^2\delta_2^1,\delta_2^1\delta_2^2\delta_2^2\delta_2^1,\delta_2^2\delta_2^1\delta_2^2\delta_2^1,\delta_2^2\delta_2^2\delta_2^2\delta_2^1,\\&\delta_2^1\delta_2^1\delta_2^1\delta_2^2,\delta_2^1\delta_2^2\delta_2^1\delta_2^2,\delta_2^2\delta_2^1\delta_2^1\delta_2^2,\delta_2^2\delta_2^2\delta_2^1\delta_2^2,\\&\delta_2^1\delta_2^1\delta_2^2\delta_2^2,\delta_2^1\delta_2^2\delta_2^2\delta_2^2,\delta_2^2\delta_2^1\delta_2^2\delta_2^2,\delta_2^2\delta_2^2\delta_2^2\delta_2^2\big]\\=&\,\delta_{16}[1,5,9,13,3,7,11,15,2,6,10,14,4,8,12,16].\end{aligned} \tag{1.4.28}$$

命题 1.4.8　设 $x_i\in\Delta_{k_i}$, $i=1,2,\cdots,n$, 则

$$W_\sigma\ltimes_{i=1}^n x_i=\ltimes_{i=1}^n x_{\sigma(i)}. \tag{1.4.29}$$

利用置换矩阵, 我们可以得到一个一般级联 Canalizing 函数的结构矩阵. 下面给出一个例子.

例 1.4.4　设 $F\in\mathcal{B}_4^{\mathcal{F}}$ 为依顺序 $\sigma=(2,4,1,3)$ 的级联 Canalizing 函数, 满足 (1.4.25), 并且

$$\begin{aligned}&a_{\sigma(1)}=a_{\sigma(4)}=1,\quad a_{\sigma(2)}=a_{\sigma(3)}=0;\\&b_{\sigma(1)}=b_{\sigma(3)}=1,\quad b_{\sigma(2)}=b_{\sigma(4)}=0.\end{aligned}$$

利用例 1.4.2 中的 M_F 及例 1.4.3 中的 W_σ, 我们知道 F 的代数状态空间表达为

$$\begin{aligned}f(x)&=M_Fx_{\sigma(1)}M_nx_{\sigma(2)}M_nx_{\sigma(3)}x_{\sigma(4)}\\&=M_F\left(I_2\otimes M_n\right)\left(I_4\otimes M_n\right)x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}\end{aligned}$$

$$
\begin{aligned}
&= M_F\left(I_2 \otimes M_n\right)\left(I_4 \otimes M_n\right) M_\sigma x_1 x_2 x_3 x_4 \\
&:= M_\sigma x,
\end{aligned} \tag{1.4.30}
$$

这里

$$
\begin{aligned}
M_\sigma &= M_F\left(I_2 \otimes M_n\right)\left(I_4 \otimes M_n\right) M_\sigma \\
&= \delta_2[1,1,2,2,1,1,1,2,1,1,*,2,1,1,1,2].
\end{aligned}
$$

第 2 章　逻辑函数与静态逻辑网络

本章讨论一般逻辑函数的一些比较深刻的性质, 包括: 逻辑函数的分解, 逻辑函数系统的隐函数存在定理, 由逻辑函数系统形成的静态逻辑网络的前提解与推论解, 静态逻辑网络的应用, 等等. 这些问题都是被长期讨论和有较多应用的二值或多值逻辑理论与方法. 本章的内容部分来自文献 [29, 108].

2.1　逻辑函数的分解

2.1.1　无重叠分解

逻辑函数的分解在逻辑线路的设计中十分重要, 它可以有效地减小布线面积、时延和能耗[58]. 因此, 这个问题从 20 世纪 50 年代开始就被广泛讨论. 例如, 关于布尔函数的分解[11], 关于多值函数的分解[79, 115], 都有许多不同情况下有效的算法. 更多的详情见 [29] 及其中的参考文献. 本章将利用矩阵半张量积方法建立一套统一的理论和算法.

定义 2.1.1　设 $F(X):\prod_{i=1}^n \mathcal{D}_i \to \mathcal{D}_0$ 给定. 记 $Y=F(X)$ (图 2.1.1 (a)).

(i) 如果存在一个分割

$$\{1,2,\cdots,n\}=\Gamma\cup\Lambda,$$

$\Gamma\cap\Lambda=\varnothing$, 以及逻辑函数

$$\begin{aligned}
&\Phi:\prod_{i\in\Gamma}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0},\\
&\Psi:\prod_{i\in\Lambda}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0},\\
&H:\mathcal{D}_{k_0}\times\mathcal{D}_{k_0}\to\mathcal{D}_{k_0},
\end{aligned}$$

使得

$$F(X)=H(\Phi(X^1),\Psi(X^2)),\tag{2.1.1}$$

这里

$$\begin{aligned}
X^1&=\{X_i\mid i\in\Gamma\},\\
X^2&=\{X_i\mid i\in\Lambda\},
\end{aligned}$$

则称 (H,Φ,Ψ) 为 F 的一个关于分割 (Γ,Λ) 的无重叠双分解 (图 2.1.1 (b)).

(ii) 如果存在一个覆盖

$$\{1,2,\cdots,n\}=\Gamma\cup\Lambda,$$

以及逻辑函数

$$\begin{aligned}&\Phi:\prod_{i\in\Gamma}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0},\\&\Psi:\prod_{i\in\Lambda}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0},\\&H:\mathcal{D}_{k_0}\times\mathcal{D}_{k_0}\to\mathcal{D}_{k_0},\end{aligned}$$

使得

$$F(X)=H(\Phi(X^1,X^2),\Psi(X^2,X^3)),\tag{2.1.2}$$

这里

$$\begin{aligned}&X^1=\{X_i\mid i\in\Gamma\backslash\{\Gamma\cap\Lambda\}\},\\&X^2=\{X_i\mid i\in\Gamma\cap\Lambda\},\\&X^3=\{X_i\mid i\in\Lambda\backslash\{\Gamma\cap\Lambda\}\},\end{aligned}$$

则称 (H,Φ,Ψ) 为 F 的一个关于覆盖 (Γ,Λ) 的重叠双分解 (图 2.1.1 (c)).

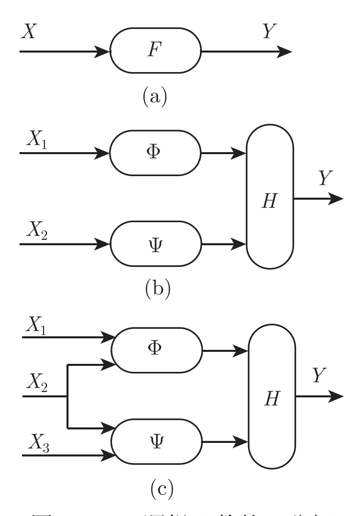


图 2.1.1　逻辑函数的双分解

为方便计, 我们设 $F(X):\prod_{i=1}^{n}\mathcal{D}_{k_i}\to\mathcal{D}_{k_0}$, 并且分割是良序的, 即

$$\begin{aligned}&\Gamma=\{1,2,\cdots,r\},\quad\Lambda=\{r+1,r+2,\cdots,r_n\},\\&p=\prod_{i=1}^{r}k_i,\quad q=\prod_{i=r+1}^{n}k_i,\end{aligned}\tag{2.1.3}$$

再将 F 的结构矩阵分块为

$$M_F = [M_1, M_2, \cdots, M_p], \tag{2.1.4}$$

这里 $M_i \in \mathcal{L}_{k_0\times q}$, $i = 1, 2, \cdots, p$. 那么, 我们有如下结果.

定理 2.1.1 设 $F(X): \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$ 满足 (2.1.3) 的假设, 记

$$T = \{T_1, T_2, \cdots, T_{k_0}\} \subset \mathcal{L}_{k_0\times k_0}, \tag{2.1.5}$$

则 F 关于 (Γ, Λ) 存在无重叠双分解, 当且仅当, 存在 T 满足 (2.1.5) 以及 $M_\Psi \in \mathcal{L}_{k_0\times q}$, 使得

$$M_i = T_{s_i} M_\Psi, \quad i = 1, 2, \cdots, p. \tag{2.1.6}$$

证明 (必要性) 假定有三个函数 H, Φ 及 Ψ, 使得 (2.1.1) 成立. 设 F 的结构矩阵 $M_F \in \mathcal{L}_{k_0\times k}$ 有 (2.1.4) 分割.

设 H 的结构矩阵

$$M_H = [H_1, H_2, \cdots, H_{k_0}],$$

这里 $H_i \in \mathcal{L}_{k_0\times k_0}$, $i = 1, \cdots, k_0$. 记 Φ 的结构矩阵

$$M_\Phi = \delta_{k_0}[i_1, i_2, \cdots, i_p].$$

Ψ 的结构矩阵 $M_\Psi \in \mathcal{L}_{k_0\times q}$. 则有

$$M_F x = M_H M_\Phi x^1 M_\Psi x^2 = M_H M_\Phi (I_p \otimes M_\Psi) x,$$

这里 $x = \ltimes_{i=1}^n x_i$, $x^1 = \ltimes_{i=1}^r x_i$, $x^2 = \ltimes_{i=r+1}^n x_i$. 由此可得

$$M_F = M_H M_\Phi (I_p \otimes M_\Psi). \tag{2.1.7}$$

先计算 $M_H M_\Phi$, 将其分割为

$$M_H M_\Phi := [N_1, N_2, \cdots, N_p].$$

直接计算可得

$$N_j = H_{i_j}, \quad j = 1, \cdots, p.$$

再记

$$M_H M_\Phi (I_p \otimes M_\Psi) := [W_1, W_2, \cdots, W_p], \tag{2.1.8}$$

则得

$$W_j = H_{i_j} M_\Psi, \quad j = 1, \cdots, p. \tag{2.1.9}$$

将 (2.1.7) 的两边分别用 (2.1.4) 及 (2.1.8)—(2.1.9) 代入, 并对比相应的块即得

$$M_i = H_{i_j} M_\Psi, \quad i = 1, \cdots, p. \tag{2.1.10}$$

定义

$$T = T_H = \{H_1, H_2, \cdots, H_{k_0}\},$$

(2.1.10) 证明了必要性.

(充分性) 设存在集合 $T = \{T_1, T_2, \cdots, T_{k_0}\}$ 和 M_Ψ 使得

$$M_i = T_{j_i} M_\Psi, \quad i = 1, \cdots, p. \tag{2.1.11}$$

令 H 的结构矩阵为

$$M_H = [T_1, T_2, \cdots, T_{k_0}],$$

以及 Φ 的结构矩阵为

$$M_\Phi = \delta_{k_0}[i_1, i_2, \cdots, i_p].$$

则直接计算即得

$$M_H M_\Phi (I_p \otimes M_\Psi) = M_F,$$

这表明

$$H(\Phi(X^1), \Psi(X^2)) = F(X). \qquad \square$$

观察方程组 (2.1.6), 不难得到以下的充分条件, 它们在求解双分解态时可大大简化求解过程.

推论 2.1.1 如果 $F(X)$ 具有关于 (Γ, Λ) 的无重叠双分解, 那么,

(i) $\{M_1, M_2, \cdots, M_p\}$ 中至多只有 k_0 个两两不等的块, 记作 $\{M_{i_1}, M_{i_2}, \cdots, M_{i_s}\}$, $s \leqslant k_0$;

(ii) 存在 s 个 $T_i \in \mathcal{L}_{k_0 \times k_0}$, 记作 $\{T_{i_1}, T_{i_2}, \cdots, T_{i_s}\}$, 使得

$$\mathrm{Col}(M_{i_k}) \subset \mathrm{Col}(T_{i_k}), \quad k = 1, 2, \cdots, s. \tag{2.1.12}$$

将定理 2.1.1 应用于布尔函数, 则有如下推论.

推论 2.1.2 考察布尔函数 $F(X) \in \mathcal{B}_n^{\mathcal{F}}$. 它具有 (Γ, Λ) 不重叠双分解, 当且仅当, 存在 $T = \{T_1, T_2\} \subset \mathcal{L}_{2\times 2}$ 和一个逻辑矩阵 $M_\Psi \in \mathcal{L}_{2\times 2^{n-r}}$, 使得 F 的结构矩阵具有如下形式

$$M_F = [\mu_1 M_\Psi, \mu_2 M_\Psi, \cdots, \mu_{2^r} M_\Psi], \tag{2.1.13}$$

这里, $\mu_i \in T$, $\forall i$.

注 2.1.1　(i) 记

$$\mathcal{L}_{2\times 2}=\{\mathcal{T}_1:=\delta_2[1,1],\ \mathcal{T}_2:=\delta_2[1,2],\mathcal{T}_3:=\delta_2[2,1],\ \mathcal{T}_4:=\delta_2[2,2]\}.$$

那么, 不管两个分量的顺序, 则可得到 10 种可能的 $T=\{\mathcal{T}_i,\mathcal{T}_j\}$, $1\leqslant i,j\leqslant 4$. 可能的 T (即函数 F 的结构矩阵) 的逻辑形式见表 2.1.1.

(ii) 再考虑关系式 (2.1.12), 需要考虑的 T 就更少了, 因此, 检验条件 (2.1.6) 就很容易了.

表 2.1.1　布尔函数 H 的可能形式

T	$\{\mathcal{T}_1,\mathcal{T}_1\}$	$\{\mathcal{T}_1,\mathcal{T}_2\}$	$\{\mathcal{T}_1,\mathcal{T}_3\}$	$\{\mathcal{T}_1,\mathcal{T}_4\}$	$\{\mathcal{T}_2,\mathcal{T}_2\}$
H	1	$\Phi\vee\Psi$	$\Phi\vee\neg\Psi$	Φ	Ψ
T	$\{\mathcal{T}_2,\mathcal{T}_3\}$	$\{\mathcal{T}_2,\mathcal{T}_4\}$	$\{\mathcal{T}_3,\mathcal{T}_3\}$	$\{\mathcal{T}_3,\mathcal{T}_4\}$	$\{\mathcal{T}_4,\mathcal{T}_4\}$
H	$\Phi\leftrightarrow\Psi$	$\Phi\wedge\Psi$	$\neg\Psi$	$\neg(\Phi\rightarrow\Psi)$	0

下面给出一个算法.

算法 2.1.1　第一步: 计算 M_F, 并将其平均分割为

$$M_F=[M_1,M_2,\cdots,M_p].$$

第二步: 检验 $\{M_1,M_2,\cdots,M_p\}$ 中有多少两两不相等的矩阵 $\{M_{i_1},M_{i_2},\cdots,M_{i_s}\}$. 如果 $s>k_0$, 则无解.

第三步: 如果 $s\leqslant k_0$, 找出 (新的) s 个不同的 $T_{i_j}\in\mathcal{L}_{k_0\times k_0}$, 满足

$$\mathrm{Col}(M_{i_j})\subset\mathrm{Col}(T_{i_j}),\quad j=1,2,\cdots,s. \tag{2.1.14}$$

如果所有可能的、不同的且满足 (2.1.14) 的 $\{T_{i_j}\}$ 都检验过, 则停止. 此时双分解不存在.

第四步: 找满足方程 (2.1.6) 的 M_Ψ. 如果存在解, 则取

$$T=[T_{i_1},T_{i_2},\cdots,T_{i_s},*],$$

这里 $*$ 任选. 如果 $M_t=M_{i_j}$, 则取

$$\mathrm{Col}_t(M_\Psi)=\delta_{k_0}^{j},\quad t=1,2,\cdots,p. \tag{2.1.15}$$

注意, 只有对每个 $j\in\{1,2,\cdots,s\}$, 得到的 M_Ψ 是相容的 (也就是说, 相等的), 这个 M_Ψ 才是有效的. 这时可构造 Φ, 其结构矩阵为

$$M_\Phi=\delta_{k_0}[s_1,s_2,\cdots,s_p], \tag{2.1.16}$$

这时, s_i 来自方程 (2.1.6).

如果无解, 回到第三步, 寻找下一组 T.

下面给一个例子.

例 2.1.1 考察一个三值函数 $F(X_1, X_2, X_3, X_4) : \mathcal{D}_3^4 \to \mathcal{D}_3$. 设其结构矩阵为

$$\begin{aligned} M_F = \delta_3[&1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,1,1,2,3,2,2,2,3,2,1,\\ &1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,1,1,2,2,2,2,2,2,2,1,\\ &1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]. \end{aligned}$$

考虑它对分割 $\Gamma = \{1,2\}$, $\Lambda = \{3,4\}$ 的双分解, 则 $p = q = 9$. 记

$$M_F = [M_1, M_2, \cdots, M_9],$$

显然, 它只有三类块

$$\begin{aligned} M^1 &= \delta_3[1,1,1,1,1,1,1,1,1],\\ M^2 &= \delta_3[1,2,2,2,2,2,2,2,1],\\ M^3 &= \delta_3[1,2,3,2,2,2,3,2,1]. \end{aligned}$$

选三个不同的 $T^i \in \mathcal{L}_{3\times 3}$, 满足 (2.1.12). 经几次试对, 不难找到

$$T^1 = \delta_3[1,1,1]; \quad T^2 = \delta_3[1,2,2]; \quad T^3 = \delta_3[1,2,3].$$

下面解方程 (2.1.6).

$$\begin{aligned} T^1 M_\Psi &= M^1,\\ T^2 M_\Psi &= M^2,\\ T^3 M_\Psi &= M^3. \end{aligned}$$

不难得到解

$$M_\Psi = \delta_3[1,2,3,2,2,2,3,2,1].$$

$$M_H = [T^1, T^2, T^3] = \delta_3[1,1,1,1,2,2,1,2,3].$$

再观察

$$M_F = [M^1, M^2, M^3, M^1, M^2, M^2, M^1, M^1, M^1].$$

$$M^i = T_i M_\Psi, \quad i = 1,2,3,$$

因此

$$M_F = \delta_3[1,2,3,1,2,2,1,1,1].$$

如果我们利用 $\{\vee, \wedge, \neg\}$ 依经典方式定义三值逻辑的逻辑算子 $\to$ 和 $\leftrightarrow$ 如下:

$$\begin{aligned} A \to B &:= (A \wedge B) \vee \neg A,\\ A \leftrightarrow B &:= (A \to B) \wedge (B \to A), \end{aligned}$$

那么, 不难发现

$$M_{\rightarrow} = M_{\Phi}, \quad M_{\leftrightarrow} = M_{\Psi}.$$

最后可得 F 的不重叠双分解为

$$F(X_1, X_2, X_3, X_4) = (X_1 \rightarrow X_2) \vee (X_3 \leftrightarrow X_4).$$

最后, 设 $\Gamma = \{i_1, i_2, \cdots, i_r\}$, $\Lambda = \{i_{r+1}, i_{r+2}, \cdots, i_n\}$. 考虑 $F(X)$ 关于分割 (Γ, Λ) 双分解. 这时, 只要把变量顺序保一个变换, 则可将分割变为良序分割, 于是, 前面关于良序分割的结论就可以用了. 于是, 我们有如下结论.

命题 2.1.1　设 $F : \prod_{i=1}^n \mathcal{D}_{k_i} \rightarrow \mathcal{D}_{k_0}$, 其结构矩阵为 M_F, $\Gamma = \{i_1, i_2, \cdots, i_r\}$, $\Lambda = \{i_{r+1}, i_{r+2}, \cdots, i_n\}$. $f(X)$ 关于分割 (Γ, Λ) 双分解可解, 当且仅当, $\tilde{F}$ 关于分割 $(\tilde{\Gamma}, \tilde{\Lambda})$ 可解. 这里

$$\begin{aligned} &\tilde{\Gamma} = \{1, 2, \cdots, r\}, \quad \tilde{\Lambda} = \{r+1, r+2, \cdots, n\}, \\ &M_{\tilde{F}} = M_f W_{\sigma}, \\ &\sigma(i_j) = j, \quad j = 1, 2, \cdots, n. \end{aligned}$$

2.1.2　重叠双分解

定义 2.1.2　设 $f : \prod_{i=1}^n \mathcal{D}_{k_i} \rightarrow \mathcal{D}_{k_0}$ 为一混合值逻辑函数, $\Gamma \cup \Theta \cup \Lambda$ 为 $\{1, 2, \cdots, n\}$ 的一个分割. F 称为具有对 $\Gamma \cup \Theta$ 和 $\Lambda \cup \Theta$ 的一个双分解, 如果存在三个逻辑函数 (H, Φ, Ψ), 其中

$$\begin{aligned} &H : \mathcal{D}_{k_0}^2 \rightarrow \mathcal{D}_{k_0}, \\ &\Phi : \prod_{i \in \Gamma \cup \Theta} \mathcal{D}_{k_i} \rightarrow \mathcal{D}_{k_0}, \\ &\Psi : \prod_{i \in \Lambda \cup \Theta} \mathcal{D}_{k_i} \rightarrow \mathcal{D}_{k_0}, \end{aligned}$$

使得

$$F(X_1, \cdots, X_n) H\left(\Phi(X^{\gamma}), \Psi(X^{\lambda})\right), \tag{2.1.17}$$

这里, $X^{\gamma} \in \prod_{i \in \Gamma \cup \Theta} \mathcal{D}_{k_i}$, $X^{\lambda} \in \prod_{i \in \Lambda \cup \Theta} \mathcal{D}_{k_i}$.

类似于无重叠的情况, 我们先假定分割是良序的. 即, 存在 $r_1 \geqslant 1$, $r_2 \geqslant 1$ 且 $r_1 + r_2 < n$, 使得

$$\begin{aligned} &\Gamma = \{1, 2, \cdots, r_1\}, \\ &\Theta = \{r_1 + 1, r_1 + 2, \cdots, r_1 + r_2\}, \\ &\Lambda = \{r_1 + r_2 + 1, r_1 + r_2 + 2, \cdots, n\}. \end{aligned} \tag{2.1.18}$$

设 $x_i = \vec{X}_i$, $i = 1, 2, \cdots, n$, 记

$$\begin{aligned}\prod_{i=1}^{r_1} k_i = p, \qquad & x^1 = \ltimes_{i=1}^{r_1} x_i,\\ \prod_{i=r_1+1}^{r_1+r_2} k_i = \ell, \qquad & x^2 = \ltimes_{i=r_1+1}^{r_1+r_2} x_i,\\ \prod_{i=r_1+r_2+1}^{n} k_i = q, \quad & x^3 = \ltimes_{i=r_1+r_2+1}^{n} x_i.\end{aligned}$$

将 F 的结构矩阵 M_F 分割成 ℓp 块如下:

$$M_F = [M_1^1, M_1^2, \cdots, M_1^\ell, \cdots, M_p^1, M_p^2, \cdots, M_p^\ell]. \tag{2.1.19}$$

关于重叠双分解有如下结论.

定理 2.1.2 设 $F : \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{k_0}$ 为一混合值逻辑函数, 其结构矩阵为 M_F. F 可依 Γ, Θ, Λ 双分解为 (2.1.17), 当且仅当,

(i) 存在一族逻辑矩阵

$$T := \{T_i \in \mathcal{L}_{k_0\times k_0} \mid i = 1, 2, \cdots, k_0\};$$

以及

(ii) 一族逻辑矩阵

$$M_\Psi^s \in \mathcal{L}_{k_0\times q}, \quad s = 1, \cdots, \ell,$$

使得 F 的结构矩阵可表示为

$$\begin{aligned} M_F = \ & [\mu_{1,1}M_\Psi^1 \quad \mu_{1,2}M_\Psi^2 \quad \cdots \quad \mu_{1,\ell}M_\Psi^\ell\\ & \ \ \mu_{2,1}M_\Psi^1 \quad \mu_{2,2}M_\Psi^2 \quad \cdots \quad \mu_{2,\ell}M_\Psi^\ell\\ & \qquad \cdots \qquad\qquad\qquad \cdots \qquad \cdots\\ & \ \ \mu_{p,1}M_\Psi^1 \quad \mu_{p,2}M_\Psi^2 \quad \cdots \quad \mu_{p,\ell}M_\Psi^\ell] \end{aligned} \tag{2.1.20}$$

这里

$$\mu_{i,j} \in T, \quad i = 1, \cdots, p, \quad j = 1, \cdots, \ell.$$

证明 (必要性) 设有三个逻辑函数 H, Φ 和 Ψ, 使 (2.1.17) 成立. 将 H 的结构矩阵等分为

$$M_H = [H_1, H_2, \cdots, H_{k_0}],$$

这里 $H_i \in \mathcal{L}_{k_0\times k_0}$, $i = 1, \cdots, k_0$.

将 Φ 的结构矩阵记作

$$M_{\Phi} = \delta_{k_0}[j_1, j_2, \cdots, j_{p\ell}].$$

将 Ψ 的结构矩阵分块表示成

$$M_{\Psi} = \left[M_{\Psi}^1, M_{\Psi}^2, \cdots, M_{\Psi}^{\ell}\right] \in \mathcal{L}_{k_0 \times q\ell},$$

这里

$$M_{\Psi}^i \in \mathcal{L}_{k_0 \times q}, \quad i = 1, \cdots, \ell.$$

定义

$$T := \{H_1, H_2, \cdots, H_{k_0}\}.$$

根据 (2.1.17) 可得

$$M_F x = M_H M_{\Phi} x^1 x^2 M_{\Psi} x^2 x^3,$$

这里 $x = \ltimes_{i=1}^n x_i$.

经矩阵-向量换位与逻辑向量降阶即得

$$M_F = M_H M_{\Phi} \left(I_{p\ell} \otimes M_{\Psi}\right) \left(I_p \otimes \mathrm{PR}_{\ell}\right). \tag{2.1.21}$$

先计算 $M_H M_{\Phi}$, 将其记作

$$M_H M_{\Phi} := [N_1, N_2, \cdots, N_{p\ell}]. \tag{2.1.22}$$

类似于无重叠的情况, 不难看出

$$N_s \in T, \quad s = 1, 2, \cdots, p\ell.$$

接下来, 我们计算 $\left(I_{p\ell} \otimes M_{\Psi}\right) \left(I_p \otimes \mathrm{PR}_{\ell}\right)$.

$$\left(I_{p\ell} \otimes M_{\Psi}\right) \left(I_p \otimes \mathrm{PR}_{\ell}\right) = I_p \otimes \left[\left(I_{\ell} \otimes M_{\Psi}\right) \mathrm{PR}_{\ell}\right]. \tag{2.1.23}$$

设法简化 $\left(I_{\ell} \otimes M_{\Psi}\right) \mathrm{PR}_{\ell}$. 注意到 $\left(I_{\ell} \otimes M_{\Psi}\right) \in \mathcal{L}_{\ell k_0 \times \ell^2 q}$, 以及 $\mathrm{PR}_{\ell} \in \mathcal{L}_{\ell^2 \times \ell}$. 将表达式从矩阵半张量积形式返回普通矩阵积形式, 则得

$$\left(I_{\ell} \otimes M_{\Psi}\right) \mathrm{PR}_{\ell} = \left(I_{\ell} \otimes M_{\Psi}\right) \left(\mathrm{PR}_{\ell} \otimes I_q\right). \tag{2.1.24}$$

将两个因子逐块写出, 则得

$$I_{\ell} \otimes M_{\Psi} = \left.\begin{bmatrix} M_{\Psi} & 0 & \cdots & 0 \\ 0 & M_{\Psi} & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & M_{\Psi} \end{bmatrix}\right\} \ell, \tag{2.1.25}$$

以及

$$\mathrm{PR}_\ell \otimes I_q = \begin{bmatrix} \left.\begin{bmatrix} I_q \\ 0 \\ \vdots \\ 0 \end{bmatrix}\right\}\ell & 0 & \cdots & 0 \\ 0 & \left.\begin{bmatrix} 0 \\ I_q \\ \vdots \\ 0 \end{bmatrix}\right\}\ell & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \left.\begin{bmatrix} 0 \\ 0 \\ \vdots \\ I_q \end{bmatrix}\right\}\ell \end{bmatrix}. \tag{2.1.26}$$

将 (2.1.25) 和 (2.1.26) 逐块相乘即得

$$(I_\ell \otimes M_\Psi)\,\mathrm{PR}_\ell = \begin{bmatrix} M_\Psi^1 & 0 & \cdots & 0 \\ 0 & M_\Psi^2 & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & M_\Psi^\ell \end{bmatrix}. \tag{2.1.27}$$

利用 (2.1.22)—(2.1.24) 以及 (2.1.27), 就可以计算 (2.1.21). 从而得到 (2.1.20). 这就证明了必要性.

(充分性) 给定

$$T = \{T_1, T_2, \cdots, T_{k_0}\},$$

则可令 H 的结构矩阵为

$$M_H = [T_1, T_2, \cdots, T_{k_0}].$$

利用

$$M_\Psi = \left[M_\Psi^1, M_\Psi^2, \cdots, M_\Psi^\ell\right]$$

即可构造 Ψ.

记

$$M_\Phi = \left[M_\Phi^{1,1}, \cdots, M_\Phi^{1,\ell}, \cdots, M_\Phi^{p,1}, \cdots, M_\Phi^{p,\ell}\right]. \tag{2.1.28}$$

因为 (2.1.20) 已知, 根据 $\mu_{\alpha,\beta}$, 我们可构造 $M_{\Phi}^{\alpha,\beta}$ 如下: 如果 $\mu_{\alpha,\beta}=T_i$, 则定义

$$M_{\Phi}^{\alpha,\beta}=\delta_{k_0}^{i}. \tag{2.1.29}$$

对于这组 (H, Φ, Ψ), 不难验证 (2.1.21). 从而保证 (2.1.17) 成立. □

由定理 2.1.2 可直接得到以下的一些必要条件.

推论 2.1.3 设 F 的结构矩阵如 (2.1.19) 所示, F 具有 (2.1.17) 形式的重叠双分解, 那么

(i) 对每个 $1\leqslant i\leqslant \ell$, 相应的矩阵集合 M_j^i, $j=1,2,\cdots,p$ 至多有 k_0 个两两不同的矩阵.

(ii) 存在 $T_j^i\in\mathcal{L}_{k_0\times k_0}$, $i=1,2,\cdots,\ell$, $j=1,2,\cdots,p$, 使得

$$\mathrm{Col}(M_j^i)\subset \mathrm{Col}(T_j^i). \tag{2.1.30}$$

并且 $T:=\{T_j^i \mid 1\leqslant i\leqslant \ell; 1\leqslant j\leqslant p\}$ 中最多有 k_0 个两两不等的矩阵.

下面给出一个算法.

算法 2.1.2 第一步: 计算 M_F, 并将其平均分割为

$$M_F=\left[M_1^1,M_1^2,\cdots,M_1^{\ell},\cdots,M_p^1,M_p^2,\cdots,M_p^{\ell}\right].$$

第二步: 对每个 $1\leqslant i\leqslant \ell$ 检验 $\{M_1^i,M_2^i,\cdots,M_p^i\}$ 中有多少两两不相等的矩阵, 如果至少有一组 i 有 $s_i>k_0$ 个两两不等的矩阵, 则无解.

第三步: 找出 (新的) 至多 k_0 个不同的 $T_j^i\in\mathcal{L}_{k_0\times k_0}$, 满足 (2.1.30). 如果所有可能的、不同的且满足 (2.1.30) 的 $\{T_{i_j}\}$ 都检验过, 则停止. 此时所要求的双分解不存在.

第四步: 找满足下列方程的 M_{Ψ}^i,

$$M_j^i=T_j^iM_{\Psi}^i,\quad 1\leqslant i\leqslant \ell;\ \ 1\leqslant j\leqslant p. \tag{2.1.31}$$

这里, $M_{\Psi}^i\in\mathcal{L}_{k_0\times q}$.

如果 (2.1.31) 的解 M_{Ψ}^i, $1\leqslant i\leqslant \ell$ 存在, 则取出 $\{T_j^i \mid 1\leqslant i\leqslant \ell;\ 1\leqslant j\leqslant p\}$ 中所有不同的矩阵. 如果所有两两不等的矩阵个数小于 k_0, 则任选其余的 $T'\in\mathcal{L}_{k_0\times k_0}$ 以补足 k_0 个, 记作 $\mathcal{T}_s$, $s=1,2,\cdots,k_0$. 依此构造

$$T=[\mathcal{T}_1,\mathcal{T}_2,\cdots,\mathcal{T}_{k_0}]. \tag{2.1.32}$$

如果 $M_t=M_j^i$, 满足 (2.1.31), 并且

$$T_j^i=\mathcal{T}_{\mu},$$

则取

$$\mathrm{Col}_t(M_\Psi)=\delta_{k_0}^{\mu}, \quad t=1,2,\cdots,p. \tag{2.1.33}$$

最后构造 Φ, 其结构矩阵如 (2.1.28) 所示. 回忆方程 (2.1.31), 设 $M_j^i=T_j^iM_\Psi$, 并且, 依集合 (2.1.32) 的序, $T_j^i=T_s$,

$$M_\Phi^{i,j}=\delta_{k_0}^{s}. \tag{2.1.34}$$

如果无解, 回到第三步, 寻找下一组 T.

下面给出一个例子.

例 2.1.2 设 $F(X):\mathcal{D}_3^4\to\mathcal{D}_3$ 为一个三值逻辑函数, 其结构矩阵为

$$\begin{aligned}M_F=\delta_3[&1,1,1,2,2,2,3,3,3,2,2,1,2,2,2,3,3,3,3,2,1,3,2,2,3,3,3,\\&1,1,1,2,2,2,3,3,3,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,\\&1,1,1,2,2,2,3,3,3,2,2,2,2,2,2,2,2,2,1,2,3,1,2,2,1,1,1].\end{aligned}$$

考虑关于 $(\Gamma\cup\Theta,\Lambda\cup\Theta)$ 的分解, 这里, $\Gamma=\{1\}$, $\Theta=\{2\}$, $\Lambda=\{3,4\}$. 于是 $p=\ell=3$, $q=9$. 将 M_F 分块为

$$M_F=[M_1^1,M_1^2,M_1^3,M_2^1,M_2^2,M_2^3,M_3^1,M_3^2,M_3^3].$$

利用算法 2.1.2, 经试错, 不难得到 $T=\{T_1,T_2,T_3\}$, 这里

$$\begin{aligned}&T_1=\delta_3[1,2,3],\\&T_2=\delta_3[2,2,2],\\&T_3=\delta_3[3,2,1].\end{aligned}$$

进而有

$$\begin{aligned}&T_1^1=T_1^2=T_1^3=T_2^1=T_3^1=T_1,\\&T_2^2=T_3^2=T_2^3=T_2,\\&T_3^3=T_3.\end{aligned}$$

于是可知, H 的结构矩阵为

$$M_H=\delta_3[1,2,3,2,2,2,3,2,1].$$

再求解以下方程

$$\begin{aligned}&T_1^1M_\Psi^1=\delta_3[1,1,1,2,2,2,3,3,3],\\&T_1^2M_\Psi^1=\delta_3[1,1,1,2,2,2,3,3,3],\end{aligned}$$

$$
\begin{aligned}
T_1^3 M_\Psi^1 &= \delta_3[1,1,1,2,2,2,3,3,3],\\
T_2^1 M_\Psi^2 &= \delta_3[2,2,1,2,2,2,3,3,3],\\
T_2^2 M_\Psi^2 &= \delta_3[2,2,2,2,2,2,2,2,2],\\
T_2^3 M_\Psi^2 &= \delta_3[2,2,2,2,2,2,2,2,2],\\
T_3^1 M_\Psi^3 &= \delta_3[3,2,1,3,2,2,3,3,3],\\
T_3^2 M_\Psi^3 &= \delta_3[2,2,2,2,2,2,2,2,2],\\
T_3^3 M_\Psi^3 &= \delta_3[1,2,3,1,2,2,1,1,1].
\end{aligned}
$$

不难解出 (其中的一组解)

$$
\begin{aligned}
M_\Psi^1 &= \delta_3[1,1,1,2,2,2,3,3,3],\\
M_\Psi^2 &= \delta_3[2,2,1,2,2,2,3,3,3],\\
M_\Psi^3 &= \delta_3[3,2,1,3,2,2,3,3,3].
\end{aligned}
$$

于是

$$
M_\Psi = \delta_3[1,1,1,2,2,2,3,3,3,2,2,1,2,2,2,3,3,3,3,2,1,3,2,2,3,3,3].
$$

再根据 (2.1.34) 找 Φ. 不难得到

$$
M_\Phi = \delta_3[1,1,1,1,2,2,1,2,3].
$$

利用 H, Ψ, Ψ 的结构矩阵, 不难得到 $F(X)$ 的二分解式

$$
F(X) = (X_1 \vee X_2) \leftrightarrow [(X_2 \vee \neg X_4) \wedge X_3].
$$

注 2.1.2 以上解法是针对良序 (Γ, Θ, Λ) 的, 如果是针对 $X_{\sigma(1)}$, $X_{\sigma(2)}, \cdots$, $X_{\sigma(n)}$ 的, 那么, 同无重叠双分解一样, 只要对结构矩阵 $M_F W_\sigma^{\mathrm{T}}$ 应用良序算法即可. 注意, $W_\sigma^{\mathrm{T}} = W_{\sigma^{-1}}$.

2.2 隐函数存在定理及其应用

2.2.1 隐函数存在定理

隐函数存在定理在数学分析中是很重要的. 但关于“存在”只有充分性结论, 没有充要条件. 试图将分析中的隐函数存在定理推广到逻辑函数, 是一种自然而合理的尝试. 不难看出, 逻辑系统的隐函数存在定理必定会在逻辑系统的分析与控制中得到类似的应用. 例如, 作为这种努力, 文献 [13] 对布尔系统给出了一个充分条件.

本节考虑 k 值逻辑函数的隐函数存在定理. 我们将通过二分解给出隐函数存在的充要条件.

考察 $r\ (<n)$ 个 k 值逻辑函数

$$G_j(X_1, X_2, \cdots, X_n) = 1, \quad j = 1, \cdots, r, \tag{2.2.1}$$

这里 $G_j : \mathcal{D}_k^n \to \mathcal{D}_k$ 为 k 值逻辑函数, 其中 $X_i \in \mathcal{D}_k$, $i = 1, \cdots, n$. 注意, (2.2.1) 右边可以是任何定常值 $c_j \in \mathcal{D}_k$, 为方便, 我们令 $c_j = 1$, 即 $x_j = \vec{X}_j = \delta_k^1$, $j = 1, 2, \cdots, n$.

记

$$\begin{aligned} x^1 &= \ltimes_{i=1}^{n-r} x_i \in \Delta_p, && \text{这里 } p = k^{n-r}, \\ x^2 &= \ltimes_{i=n-r+1}^{n} x_i \in \Delta_q, && \text{这里 } q = k^r. \end{aligned}$$

令 $G = (G_1, \cdots, G_r)$, 则 $G : \mathcal{D}_k^n \to \mathcal{D}_k^r$, 其向量形式则为 $g : \Delta_p \times \Delta_q \to \Delta_q$. 那么, (2.2.1) 可表示为代数状态空间表达式

$$M_G x^1 x^2 = \delta_q^1, \tag{2.2.2}$$

这里, M_G 是 G 的结构矩阵.

定义 2.2.1 如果存在唯一的一组 k 值逻辑函数族 $\Phi_j : \mathcal{D}_k^{n-r} \to \mathcal{D}_k$, $j = n-r+1, n-r+2, \cdots, n$, 使得当

$$X_j = \Phi_j(X_1, \cdots, X_{n-r}), \quad j = n-r+1, \cdots, n \tag{2.2.3}$$

时 (2.2.1) 成立, 则 (2.2.3) 称为由 (2.2.1) 所决定的隐函数.

隐函数存在定理要求回答, 当 (2.2.1) 满足什么条件时, 存在唯一函数族 $\{\Phi_j \mid j = n-r+1, \cdots, n\}$, 使得 (2.2.3) 满足 (2.2.1).

定义一个逻辑算子 $\leftrightarrow : \mathcal{D}_q \times \mathcal{D}_q \to \mathcal{D}_q$ 如下:

$$\xi \leftrightarrow \eta = \begin{cases} \mathbf{1}_q, & X^1 = X^2, \\ \zeta, & X^1 \neq X^2, \end{cases} \tag{2.2.4}$$

这里 $\zeta \in \mathcal{D}_q$ 并且 $\zeta \neq \mathbf{1}_q$.

那么, (2.2.3) 可以表示成

$$F_{\leftrightarrow}(\Phi(X^1), X^2) := \Phi(X^1) \leftrightarrow X^2 = \mathbf{1}_q. \tag{2.2.5}$$

利用这个记号可知, (2.2.1) 与 (2.2.3) 等价, 当且仅当,

$$G(X^1, X^2) = F_{\leftrightarrow}(\Phi(X^1), X^2). \tag{2.2.6}$$

注意到在定义式 (2.2.4) 中 $\zeta \neq \mathbf{1}_q$ 可以是任意常值, 因此, 在 (2.2.6) 中可以构造许多不同的 $F_{\leftrightarrow}$. 我们构造所有可能的 F 的结构矩阵 M_F.

定义 Ξ_i, $1 \leqslant i \leqslant q$ 如下:

$$\Xi_i = \left\{ E_i \in \mathcal{L}_{q\times q} \middle| \operatorname{Col}_i(E_i) = \delta_q^1;\ \operatorname{Col}_j(E_i) \neq \delta_q^1,\ j \neq i \right\}. \tag{2.2.7}$$

利用 Ξ_i, 我们构造一组 $\{E_i\}$ 集合:

$$\mathcal{E}_q := \left\{ \{E_1, E_2, \cdots, E_q\} \middle| E_i \in \Xi_i,\ i = 1, 2, \cdots, q \right\}. \tag{2.2.8}$$

再利用每一组 $T \in \mathcal{E}_q$ 构造 $F : \mathcal{D}_q \times \mathcal{D}_q \to \mathcal{D}_q$ 的结构矩阵

$$M_F = E_T := [E_1, E_2, \cdots, E_q].$$

下面的引理说明, 这样的 E_T 的确是相应的结构矩阵.

引理 2.2.1　设 ξ, $\eta \in \mathcal{D}_q$ 为两个 q 值逻辑变量, $x = \vec{\xi}$, $y = \vec{\eta}$. 那么, $\xi = \eta$, 当且仅当, 存在一个 $T \in \mathcal{E}_q$ 使得

$$E_T x y = \delta_q^1. \tag{2.2.9}$$

证明　记

$$E_T = [E_1, E_2, \cdots, E_q],$$

并设 $x = \delta_q^\alpha$, $y = \delta_q^\beta$. 直接计算可得

$$E_T x y = \operatorname{Col}_\beta(E_\alpha).$$

因此, (2.2.9) 成立, 当且仅当, $\alpha = \beta$. □

从以上讨论可得如下定理.

定理 2.2.1 (隐函数存在定理)　考察 $G = (G_1, G_2, \cdots, G_r)$ 如 (2.2.1). 记 G 的结构矩阵为 $M_G = [M_1, M_2, \cdots, M_q]$. 则存在唯一一组如 (2.2.3) 所示的隐函数 X_j, $j = n-r+1, n-r+2, \cdots, n$, 满足 (2.2.1), 当且仅当, 存在

$$T = \{E_1, E_2, \cdots, E_q\} \in \mathcal{E}_q,$$

使得

$$M_i \in T, \quad i = 1, \cdots, q. \tag{2.2.10}$$

证明　将 (2.2.3) 表示为其代数状态空间表达式

$$x^2 = M_\phi x^1.$$

根据引理 2.2.1, (2.2.1) 可表示为 (2.2.3), 当且仅当, 存在 $T \in \mathcal{E}_q$ 使得 F 满足 $M_F = E_T$ 并且

$$M_F M_\phi x^1 x^2 = \delta_q^1, \tag{2.2.11}$$

这里, M_F 是 F 的结构矩阵. 比较等式 (2.2.11) 与 (2.2.2), 即得

$$M_G x^1 x^2 = M_F M_\phi x^1 x^2, \tag{2.2.12}$$

这里 $M_F = T$, 而 $T \in \mathcal{E}_q$.

现在可以形式地将 x^2 看作 $x^2 = M_\psi x^2$ 且 $M_\psi = I_q$, 根据引理 2.2.1 和定理 2.1.1, 结论显见. □

2.2.2 奇异布尔网络

作为隐函数存在定理的应用, 我们讨论奇异布尔网络. 奇异布尔网络包括静态与动态两个部分: 静态部分由一组布尔函数决定, 动态部分由一组布尔差分方程决定. 这类系统大量存在, 在连续时间动态系统中, 它称为代数-微分方程.

一个奇异布尔网络可描述如下:

$$X_i(t+1) = F_i(X_1(t), X_2(t)\cdots, X_n(t)), \quad i = 1, \cdots, n-r, \tag{2.2.13a}$$

$$\xi_j = G_j(X_1(t), X_2(t), \cdots, X_n(t)), \quad j = n-r+1, \cdots, n, \tag{2.2.13b}$$

这里, $X_i(t) \in \mathcal{D}$, $i = 1, \cdots, n$ 为逻辑变量, $\xi_j \in \mathcal{D}$, $j = n-r+1, \cdots, n$ 为逻辑常数.

显然, 如果能从 (2.2.13b) 中解出 X_j, $j = n-r+1, \cdots, n$, 即得到

$$X_j(t) = \varphi_j(X_1, X_2, \cdots, X_{n-r}), \quad j = n-r+1, \cdots, n. \tag{2.2.14}$$

那么, 将 (2.2.14) 代入 (2.2.13a), 则可得到标准的布尔网络方程. 这是解代数-微分 (差分) 方程最直观的方法. 而解 (2.2.14) 是否存在则依赖于隐函数存在定理.

下面给出一个例子.

例 2.2.1 考察如下的奇异布尔网络:

$$\begin{cases} X_1(t+1) = X_2(t) \to X_4(t), \\ X_2(t+1) = X_1(t) \wedge X_3(t), \\ 1 = (X_3(t) \bar{\vee} X_4(t)) \leftrightarrow (X_1(t) \bar{\vee} X_2(t)), \\ 0 = X_4(t) \bar{\vee} (X_1(t) \vee X_2(t)). \end{cases} \tag{2.2.15}$$

我们希望从后两个方程中解出 X_3 和 X_4. 首先, 将它们转化为标准形式:

$$\begin{cases} G_1(X_1,X_2,X_3,X_4) := (X_3(t)\bar{\vee}X_4(t)) \leftrightarrow (X_1(t)\bar{\vee}X_2(t)) = 1, \\ G_2(X_1,X_2,X_3,X_4) := X_4(t) \leftrightarrow (X_1(t)\vee X_2(t)) = 1, \end{cases} \tag{2.2.16}$$

这里, G_2 由 (2.2.15) 的最后一个方程两边取“非”而得.

记 $x_i = \vec{X}_i$, $i=1,2,3,4$, $x = \ltimes_{i=1}^4 x_i$, 则得

$$\begin{cases} M_{G_1}x = \delta_2[1,2,2,1,2,1,1,2,2,1,1,2,1,2,2,1]x, \\ M_{G_2}x = \delta_2[1,2,1,2,1,2,1,2,1,2,1,2,2,1,2,1]x. \end{cases} \tag{2.2.17}$$

于是, $G=(G_1,G_2)$ 的结构矩阵为

$$\begin{aligned} M_G &= M_{G_1} * M_{G_2} \\ &= \delta_4[1,4,3,2,3,2,1,4,3,2,1,4,2,3,4,1] \\ &:= [M_1,M_2,M_3,M_4]. \end{aligned} \tag{2.2.18}$$

利用隐函数存在定理 (定理 2.2.1) 不难检验, 从 G 中可唯一解出 $X_3(t)$ 及 $X_4(t)$. 利用算法 2.1.1 可得如下解:

$$T = \{E_1,E_2,E_3,E_4\} \in \mathcal{K}_{4\times 16},$$

这里

$$\begin{aligned} E_1 &= \delta_4[1,4,3,2], \\ E_3 &= \delta_4[3,2,1,4], \\ E_4 &= \delta_4[2,3,4,1], \end{aligned}$$

以及 E_2 任取. 进而可解得

$$M_\Phi = \delta_4[1,3,3,4],$$

即

$$x_3(t)x_4(t) = \delta_4[1,3,3,4]x_1(t)x_2(t).$$

于是

$$\begin{aligned} x_3(t) &= (I_2\otimes \mathbf{1}_2)\,x_3(t)x_4(t) = \delta_2[1,2,2,2]x_1(t)x_2(t), \\ x_4(t) &= (\mathbf{1}_2\otimes I_2)\,x_3(t)x_4(t) = \delta_2[1,1,1,2]x_1(t)x_2(t). \end{aligned} \tag{2.2.19}$$

返回逻辑表达式, 则得

$$\begin{cases} X_3(t) = X_1(t)\wedge X_2(t), \\ X_4(t) = X_1(t)\vee X_2(t). \end{cases} \tag{2.2.20}$$

将 (2.2.20) 代回 (2.2.13a), 可得到标准布尔网络

$$\begin{cases} X_1(t+1) = X_2(t) \to (X_1(t) \vee X_2(t)), \\ X_2(t+1) = X_1(t) \wedge X_2(t). \end{cases} \tag{2.2.21}$$

(2.2.20)—(2.2.21) 与奇异布尔网络 (2.2.13) 等价. 我们可以从 (2.2.21) 求出解 $(X_1(t),\ X_2(t))$, 再代入 (2.2.20) 求出 $(X_3(t),\ X_4(t))$.

注 2.2.1 (i) 上述方法对奇异 k 逻辑网络同样可用.

(ii) 混合值逻辑系统的隐函数存在定理原则上也可以类似地求得, 但在表述和应用上都比较复杂, 留待有兴趣的读者深入探讨.

2.3 静态逻辑网络的 Ledley 解

2.3.1 分割与真值矩阵

给定一组布尔函数方程

$$\Sigma: \begin{cases} \varphi_1(X_1, X_2, \cdots, X_n) = c_1, \\ \cdots\cdots \\ \varphi_s(X_1, X_2, \cdots, X_n) = c_s, \end{cases} \tag{2.3.1}$$

这里 $X_j \in \mathcal{D} := \{0,1\}$, $j = 1, \cdots, n$ 为布尔变量, $c_i \in \mathcal{D}$, $i = 1, \cdots, s$ 为布尔常数, $\varphi_i : \mathcal{D}^n \to \mathcal{D}$, $i = 1, \cdots, s$ 为布尔函数.

隐函数存在定理讨论在这组方程中是否指定的一组布尔变量可以表示成其余变量的布尔函数. 那么, 粗略地说, 把变量任意地分成两组, 看其中的一组是否为另一组的函数, 就是隐函数存在定理的一种推广. 最早讨论这个问题的是 Ledley [82]. 文献 [9] 对 Ledley 理论及其应用做了较全面的论述. 文献 [108] 利用矩阵半张量积方法系统讨论了这个问题, 它是本节内容的主要来源.

首先, 考虑变量的一个分割

$$\Gamma = X \cup Y, \tag{2.3.2}$$

这里,

$$\begin{aligned} &\Gamma := \{X_1, X_2, \cdots, X_n\}, \\ &X := \{\xi_1, \xi_2, \cdots, \xi_p\}, \\ &Y := \{\eta_1, \eta_2, \cdots, \eta_q\}, \quad p + q = n. \end{aligned} \tag{2.3.3}$$

简称分割 $\Gamma = X \cup Y$ 为分割 X–Y.

我们寻找解

$$\eta_\ell = F_\ell(\xi_1, \xi_2, \cdots, \xi_p), \quad \ell = 1, \cdots, q. \tag{2.3.4}$$

解 (2.3.4) 称为基于分割 $X\!-\!Y$ 的解.

定义 2.3.1　考察网络 (2.3.1) (即, Σ) 以及分割 (2.3.2) (即, 分割 $X\!-\!Y$). 记 $x_i = \vec{\xi}_i$, $i = 1, 2, \cdots, p$, $y_j = \vec{\eta}_j$, $j = 1, 2, \cdots, q$. 一个矩阵

$$T_\Sigma^{(X,Y)} = (t_{ij}) \in \mathcal{B}_{2^q \times 2^p}$$

称为网络 (2.3.1) 关于分割 $X\!-\!Y$ 的真值矩阵, 如果

$$t_{i,j} = \begin{cases} 1, & x = \delta_{2^p}^j,\ y = \delta_{2^q}^i \text{ 使 (2.3.1) 成立}, \\ 0, & \text{其他}. \end{cases} \tag{2.3.5}$$

下面给出一个真值矩阵构造法.

算法 2.3.1　第一步: 将 (2.3.1) 变为

$$\varphi(\xi_1, \cdots, \xi_n) = 1. \tag{2.3.6}$$

为此目的, 首先将所有 c_i, $i = 1, \cdots, s$ 变为 1. 即令

$$\begin{cases} \tilde{\varphi}_i = \varphi_i, & c_i = 1, \\ \tilde{\varphi}_i \neq \varphi_i, & c_i = 0. \end{cases}$$

再设

$$\varphi := \bigwedge_{i=1}^{s} \tilde{\varphi}_i.$$

第二步: 设 $X = (X_1, \cdots, X_p)$ 及 $Y = (X_{p+1}, \cdots, X_n)$. (这时可能要重排变量顺序.) 设在这个顺序下 φ 的真值表为 $T \in \mathcal{D}^{2^n}$. 那么, 将 T 均分为 2^s 块, 即

$$T = \begin{bmatrix} T_1 \\ \vdots \\ T_{2^s} \end{bmatrix},$$

这里 $T_i \in \mathcal{D}^{2^{n-p}}$, $i = 1, \cdots, 2^p$.

第三步:

$$T_{(2.3.1)}^{(X,Y)} = [T_1, T_2, \cdots, T_{2^p}]. \tag{2.3.7}$$

下面给出一个例子.

例 2.3.1　考察网络

$$\Sigma: \ \varphi(X_1, X_2, X_3) = (X_1 \wedge X_2) \vee (\neg X_3) = 1. \tag{2.3.8}$$

φ 的真值见表 2.3.1.

表 2.3.1　(2.3.8) 的真值表

X_1	X_2	X_3	$X_1 \wedge X_2$	$\neg X_3$	$\varphi(X_1, X_2, X_3)$
1	1	1	1	0	1
1	1	0	1	1	1
1	0	1	0	0	0
1	0	0	0	1	1
0	1	1	0	0	0
0	1	0	0	1	1
0	0	1	0	0	0
0	0	0	0	1	1

定义分割

$$\Gamma = \{X_1, X_2, X_3\} = X^i \cup Y^i, \quad i = 1, 2, 3, 4, 5, 6. \tag{2.3.9}$$

(i) 设 $X^1 := \{X_1, X_2\}$ 及 $Y^1 = \{X_3\}$. 利用算法 2.3.1, 可得表 2.3.2.

表 2.3.2　(2.3.8) 关于 $\boldsymbol{X^1}$−$\boldsymbol{Y^1}$ 的真值矩阵

Y^1	X^1			
	1, 1	1, 0	0, 1	0, 0
1	1	0	0	0
0	1	1	1	1

写成矩阵形式, 则得

$$T_{(2.3.8)}^{(X^1, Y^1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

(ii) 如果 $X^2 = \{X_2, X_3\}$, $Y^2 = \{X_1\}$, 将变量重排为 $(X_2,\ X_3,\ X_1)$, 从对应的真值表可得

$$T_{(2.3.8)}^{(X^2, Y^2)} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

类似地, 我们可得

(iii) 当 $X^3 = \{X_1, X_3\}$, $Y^3 = \{X_2\}$ 时, 有

$$T_{(2.3.8)}^{(X^3, Y^3)} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

(iv) 当 $X^4=\{X_1\}, Y^4=\{X_2,X_3\}$ 时, 有

$$T_{(2.3.8)}^{(X^4,Y^4)}=\begin{bmatrix}1&0\\1&1\\0&0\\1&1\end{bmatrix}.$$

(v) 当 $X^5=\{X_2\}, Y^5=\{X_1,X_3\}$ 时, 有

$$T_{(2.3.8)}^{(X^5,Y^5)}=\begin{bmatrix}1&0\\1&1\\0&0\\1&1\end{bmatrix}.$$

(vi) 当 $X^6=\{X_3\}, Y^6=\{X_1,X_2\}$ 时, 有

$$T_{(2.3.8)}^{(X_6,Y_6)}=\begin{bmatrix}1&1\\0&1\\0&1\\0&1\end{bmatrix}.$$

注 2.3.1　(i) 在每一个分块内部 (即在 X 或 Y 内部), 我们要求变量按原来的顺序排列, 否则, 会出现列或行的元素的置换, 从而影响了真值矩阵的唯一性.

(ii) 由算法 2.3.1 可见, 真值矩阵只是真值表的重排, 因此, 它包括了原系统的全部信息. 这就是为什么我们可以从它出发, 对系统进行分析与设计.

2.3.2　Ledley 的前提解与推论解

定义 2.3.2　考察由 (2.3.1) 决定的逻辑关系. 设存在一组分割如 (2.3.3), 以及基于这组分割的一组函数如 (2.3.4).

(i) (2.3.4) 称为 (2.3.1) 的前提解 (antecedence solution), 如果 (2.3.4) $\Rightarrow$ (2.3.1). 即满足 (2.3.4) 的逻辑变量也满足 (2.3.1).

(ii) (2.3.4) 称为 (2.3.1) 的推论解 (consequence solution), 如果 (2.3.1) $\Rightarrow$ (2.3.4). 即满足 (2.3.1) 的逻辑变量也满足 (2.3.4).

先考虑前提解, 我们从一个实例入手.

例 2.3.2 [9]　考察一个逻辑网络

$$\varphi(X_1,X_2,X_3)=(\neg X_1\wedge X_3)\vee(X_1\wedge X_2\wedge\neg X_3)=0. \tag{2.3.10}$$

任选一分割如下: $X=\{X_1,X_2\}$, $Y=\{X_3\}$. 那么, 不难算出其对应分割 $X-Y$ 真值见表 2.3.3 所示.

$$T_{(2.3.10)}^{(X,Y)}=\begin{bmatrix}1&1&0&0\\0&1&1&1\end{bmatrix}. \tag{2.3.11}$$

表 2.3.3 (2.3.10) 关于 $X-Y$ 的真值矩阵

Y	X			
	1, 1	1, 0	0, 1	0, 0
1	1	1	0	0
0	0	1	1	1

下面我们寻找

$$X_3=F(X_1,X_2) \tag{2.3.12}$$

形式的解. 设其结构矩阵 $M_F\in\mathcal{L}_{2\times 4}$ 满足

$$M_F\leqslant T_{(2.3.10)}^{(X,Y)}. \tag{2.3.13}$$

注意, 设 $A=(a_{i,j})$, $B=(b_{i,j})\in\mathcal{M}_{m\times n}$, $A\leqslant B$ 表示

$$a_{i,j}\leqslant b_{i,j},\quad i=1,\cdots,m,\quad j=1,\cdots,n.$$

不难验证, 如果 (2.3.13) 成立, 则 (2.3.12) 是 (2.3.10) 的前提解. 因为使 (2.3.12) 成立的自变量值对应于 M_F 中的 1, 而这时 (2.3.10) 的真值矩阵的相应值也是 1, 这说明这组自变量值也保证了 (2.3.10) 成立.

回到 (2.3.10), 不难看出, 满足 (2.3.13) 的结构矩阵有两个, 它们是

$$M_1=\delta_2[1,1,2,2];\quad M_2=\delta_2[1,2,2,2].$$

因此, 对应于分割 $X-Y$, 有两个前提解, 它们分别为

$$x_3=\delta_2[1,1,2,2]x_1x_2\quad\to\quad X_3=X_1 \tag{2.3.14}$$

和

$$x_3=\delta_2[1,2,2,2]x_1x_2\quad\to\quad X_3=X_1\wedge X_2. \tag{2.3.15}$$

为表述一般结果, 我们回忆布尔向量的汉明权重: 设 $X=(x_1,\cdots,x_n)\in\mathcal{B}_n$ 为一布尔向量, 则其汉明权重为

$$w_H(X)=\sum_{i=1}^{n}x_i. \tag{2.3.16}$$

根据例 2.3.2 的讨论, 不难推得以下的结论.

定理 2.3.1 考察静态布尔网络 (2.3.10) (或曰 Σ).

(i) 设 $X-Y$ 为 $\Gamma=\{X_1,X_2,\cdots,X_n\}$ 的一个分割. $|X|=p>0, |Y|=q>0$. 对应于 $X-Y$, 真值矩阵为

$$T:=T_{\Sigma}^{(X,Y)}\in\mathcal{B}_{2^q\times 2^p}.$$

那么, (2.3.10) (或曰 Σ) 具有前提解, 当且仅当, T 的每一列的汉明权重均不为零. 也就是说,

$$w_j:=w_H(\mathrm{Col}_j(T))>0,\quad j=1,\cdots,2^p. \tag{2.3.17}$$

(ii) 设存在一个逻辑矩阵 $M\in\mathcal{L}_{2^q\times 2^p}$ 满足 $M\leqslant T$, 那么, $\eta=F(\xi)$, 这里 $M_F=M$, 就是 (2.3.10) (或曰 Σ) 的前提解.

(iii) 对应于这组分割, 网络 (2.3.10) 共有

$$r=\prod_{j=1}^{2^p}w_j \tag{2.3.18}$$

个前提解.

(iv) 设 X^i-Y^i, $i=1,\cdots,\ell$ 为 Γ 所有可能的分割, $|X^i|=p_i$. 网络 (2.3.10) 所有可能的前提解个数, 记作 s_r, 满足

$$s_r\leqslant\sum_{i=1}^{\ell}\prod_{j=1}^{2^{p_i}}w_H\left[\mathrm{Col}_j\left(T_{\Sigma}^{(X_i,Y_i)}\right)\right]. \tag{2.3.19}$$

注 2.3.2 (i) 不像通常的分割, 在讨论 Ledley 解时, 变量集 Γ 的分割 $X-Y$ 是有序的, 它们的位置不能交换, 这是因为它们在函数表示中有一组 (X) 是自变量, 有一组 (Y) 是因变量, 作用不同.

(ii) (2.3.19) 之所以成不等式是因为不同的分割可能出现同样的前提解. 后面的例子 (见例 2.3.3) 中就有这种情况.

例 2.3.3 回忆例 2.3.1. 那里有六个不同的分割.

分割一: $T_{(2.3.8)}^{(X^1,Y^1)}$, 这时有两个前提解:

$$\begin{aligned}
x_3=\delta_2[1,2,2,2]x_1x_2 &\quad\Rightarrow\quad X_3=X_1\wedge X_2;\\
x_3=\delta_2[2,2,2,2]x_1x_2 &\quad\Rightarrow\quad X_3=0.
\end{aligned}$$

分割二: $T_{(2.3.8)}^{(X^2,Y^2)}$, 这时 $w_H\left(\mathrm{Col}_3(T_{(2.3.8)}^{(X_2,Y_2)})\right)=0$, 故没有前提解.

分割三: $T_{(2.3.8)}^{(X^3,Y^3)}$, 这时 $w_H\left(\mathrm{Col}_3(T_{(2.3.8)}^{(X_3,Y_3)})\right)=0$, 故没有前提解.

分割四: $T_{(2.3.8)}^{(X^4,Y^4)}$, 这时有 6 个前提解.

(i) $x_2x_3=\delta_4[1,2]x_1$: 于是有

$$x_2=\left(I_2\otimes\mathbf{1}_2^{\mathrm{T}}\right)\delta_4[1,2]x_1=\delta_2[1,1]\xi_1=\delta_2^1,$$
$$x_3=\left(\mathbf{1}_2^{\mathrm{T}}\otimes I_2\right)\delta_4[1,2]x_1=\delta_2[1,2]\xi_1=x_1,$$

故前提解为

$$\begin{cases}X_2=1,\\X_3=X_1.\end{cases}$$

类似地, 我们有

(ii) $x_2x_3=\delta_4[1,4]x_1$: 前提解为

$$\begin{cases}X_2=X_1,\\X_3=X_1.\end{cases}$$

(iii) $x_2x_3=\delta_4[2,2]x_1$: 前提解为

$$\begin{cases}X_2=1,\\X_3=0.\end{cases}$$

(iv) $x_2x_3=\delta_4[2,4]x_1$: 前提解为

$$\begin{cases}X_2=X_1,\\X_3=0.\end{cases}$$

(v) $x_2x_3=\delta_4[4,2]x_1$: 前提解为

$$\begin{cases}X_2=\neg X_1,\\X_3=0.\end{cases}$$

(vi) $x_2x_3=\delta_4[4,4]x_1$: 前提解为

$$\begin{cases}X_2=0,\\X_3=0.\end{cases}$$

分割五: $T_{(2.3.8)}^{(X^5,Y^5)}$, 这时有 6 个前提解.

(i) $x_1x_3=\delta_4[1,2]x_2$: 前提解为

$$\begin{cases}X_1=1,\\X_3=X_2.\end{cases}$$

(ii) $x_1x_3 = \delta_4[1,4]x_2$: 前提解为

$$\begin{cases} X_1 = X_2, \\ X_3 = X_2. \end{cases}$$

(iii) $x_1x_3 = \delta_4[2,2]x_2$: 前提解为

$$\begin{cases} X_1 = 1, \\ X_3 = 0. \end{cases}$$

(iv) $x_1x_3 = \delta_4[2,4]x_2$: 前提解为

$$\begin{cases} X_1 = X_2, \\ X_3 = 0. \end{cases}$$

(v) $x_1x_3 = \delta_4[4,2]x_2$: 前提解为

$$\begin{cases} X_1 = \neg X_2, \\ X_3 = 0. \end{cases}$$

(vi) $x_1x_3 = \delta_4[4,4]x_2$: 前提解为

$$\begin{cases} X_1 = 0, \\ X_3 = 0. \end{cases}$$

分割六: $T_{(2.3.8)}^{(X^6,Y^6)}$, 这时有 4 个前提解.
(i) $x_1x_2 = \delta_4[1,1]x_3$: 前提解为

$$\begin{cases} X_1 = 1, \\ X_2 = 1. \end{cases}$$

(ii) $x_1x_2 = \delta_4[1,2]x_3$: 前提解为

$$\begin{cases} X_1 = 1, \\ X_2 = X_3. \end{cases}$$

(iii) $x_1x_2 = \delta_4[1,3]x_3$: 前提解为

$$\begin{cases} X_1 = X_3, \\ X_2 = 1. \end{cases}$$

(iv) $x_1x_2 = \delta_4[1,4]x_3$: 前提解为

$$\begin{cases} X_1 = X_3, \\ X_2 = X_3. \end{cases}$$

值得注意的是, 分割五中的 (i) 与分割六中的 (ii) 是一样的. 这就是注 2.3.2 中指出的可能重复的情况.

注 2.3.3 从上面例子不难看出

$$T_\Sigma^{(X,Y)} = \left[T_\Sigma^{(Y,X)}\right]^{\mathrm{T}}.$$

但这个关系式与换序分割导出的前提解无关, 因为 X, Y 在函数表达中的作用完全不同.

回到网络 (2.3.10) , 下面讨论推论解. 类似于前提解的讨论, 不难得到如下结论.

定理 2.3.2 考察网络 (2.3.10) (或曰 Σ).

(i) 设 $X\text{–}Y$ 为 $\Gamma = \{X_1, X_2, \cdots, X_n\}$ 的一个分割. $|X| = p > 0, |Y| = q > 0$. 对应于分割 $X\text{–}Y$, 网络 Σ 的真值矩阵为 $T := T_\Sigma^{(X,Y)} \in \mathcal{B}_{2^q \times 2^p}$. 那么, (2.3.10) 有推论解, 当且仅当, T 的每一列的汉明指数不大于 1, 即

$$w_j := w_H(\mathrm{Col}_j(T)) \leqslant 1, \quad j = 1, \cdots, 2^p. \tag{2.3.20}$$

(ii) 形如 (2.3.4) 的函数是 (2.3.10) 的推论解, 当且仅当, (2.3.4) 的结构矩阵 $M_F \in \mathcal{L}_{2^q \times 2^p}$ 满足

$$T \leqslant M_F. \tag{2.3.21}$$

(iii) 对于给定的分割 $X\text{–}Y$ 如上, 则有

$$r = 2^{\mu p} \tag{2.3.22}$$

个推论解, 这里

$$\mu = \left|\{\mathrm{Col}_k(T) \in \mathrm{Col}(T) | \, \mathrm{Col}_k(T) = 0\}\right|.$$

(iv) 设 (X_i, Y_i), $i = 1, \cdots, \ell$ 为 Γ 的所有可能的分割, 其相应的推论解为 r_i 个, 则 Σ 所有推论解的数目为

$$s_r \leqslant \sum_{j=1}^{\ell} r_i. \tag{2.3.23}$$

下面考虑几个例子.

例 2.3.4　回忆例 2.3.3 (即例 2.3.1). 根据定理 2.3.2, 网络 (2.3.8) 没有推论解.

例 2.3.5　考察网络

$$\varphi(X_1, X_2, X_3) = [X_1 \wedge (X_2 \leftrightarrow X_3)] \vee [\neg X_1 \wedge (\neg X_2 \wedge X_3)] = 1. \tag{2.3.24}$$

(i) 设 $X^1 = \{X_1, X_2\}$, $Y^1 = \{X_3\}$, (2.3.24) 关于 X^1–Y^1 的真值矩阵为

$$T_{(2.3.24)}^{(X^1,Y^1)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{2.3.25}$$

根据定理 2.3.2, 这里有两个推论解:

$$x_3 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} x_1 x_2,$$

以及

$$x_3 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} x_1 x_2.$$

回到逻辑形式, 则得

$$X_3 = X_1 \to X_2, \tag{2.3.26}$$

以及

$$X_3 = X_1 \leftrightarrow X_2. \tag{2.3.27}$$

(ii) 设 $X^2 = \{X_1, X_3\}$, $Y^2 = \{X_2\}$, (2.3.24) 关于 X^2–Y^2 的真值矩阵为

$$T_{(2.3.24)}^{(X^2,Y^2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}. \tag{2.3.28}$$

根据定理 2.3.2, 这里有两个推论解:

$$X_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} X_1 X_3,$$

以及

$$X_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} X_1 X_3.$$

回到逻辑形式, 则得

$$X_2 = X_1 \leftrightarrow X_3, \tag{2.3.29}$$

以及

$$X_2 = X_1 \wedge X_3. \tag{2.3.30}$$

(iii) 设 $X^3 = \{X_2, X_3\}$, $Y^3 = \{X_1\}$, (2.3.24) 关于 X^3–Y^3 的真值矩阵为

$$T_{(2.3.24)}^{(X^3,Y^3)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.3.31}$$

根据定理 2.3.2, 这里有两个推论解:

$$x_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} x_2 x_3,$$

以及

$$x_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} x_2 x_3.$$

回到逻辑形式, 则得

$$X_1 = X_2 \wedge \neg X_3, \tag{2.3.32}$$

以及

$$X_1 = X_2 \leftrightarrow X_3. \tag{2.3.33}$$

注意,

$$T_{\varphi}^{(X,Y)} = \left(T_{\varphi}^{(Y,X)}\right)^{\mathrm{T}}, \tag{2.3.34}$$

于是对分割 X–Y, 当 $|X| = 1$, $|Y| = 2$, 时, 其结构矩阵满足 $M_F \leqslant T$ 的布尔函数 F 不存在. 因此, 对 (2.3.24), 关于分割 X–Y, $|X| = 1$, $|Y| = 2$ 的推论解不存在.

2.3.3 混合值逻辑网络的 Ledley 解

前面的关于布尔网络 Ledley 解的讨论与结论均可以平行地推广到 k 值和混合值逻辑网络.

考察混合值逻辑系统

$$\Sigma: \quad \begin{cases} \varphi_1(X_1, X_2, \cdots, X_n) = c_1, \\ \qquad\qquad \cdots\cdots \\ \varphi_s(X_1, X_2, \cdots, X_n) = c_s, \end{cases} \tag{2.3.35}$$

这里, $X_i \in \mathcal{D}_{k_i}$, $i = 1, \cdots, n$, $\varphi_j : \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{\mu_j}$, $c_j \in \mathcal{D}_{\mu_j}$, $j = 1, \cdots, s$. 设 $X{-}Y$ 为 $\Gamma\{X_1, X_2, \cdots, X_n\}$ 的一个分割,

$$\alpha := \prod_{\{i|X_i \in X\}} k_i; \quad \beta := \prod_{\{i|X_i \in Y\}} k_i. \tag{2.3.36}$$

那么, 类似于布尔网络的情况, 我们就可以定义一个真值矩阵

$$T_{\Sigma}^{(X,Y)} \in \mathcal{B}_{\alpha \times \beta}. \tag{2.3.37}$$

注意到, 虽然逻辑变量取的是逻辑值, 例如, 在三值逻辑中, 逻辑变量可取 0.5, 但真值矩阵却是一个布尔矩阵, 这是因为, 一组逻辑变量值使网络方程 (2.3.35) 或者成立, 或者不成立. 于是, 相应的真值矩阵的元素或许为 1, 或许为 0.

对于混合值逻辑网络, 我们通常从其代数状态空间表达式出发进行讨论, 以避免逻辑系统-向量形式之间麻烦而又乏味的转换.

记网络 (2.3.35) 的代数状态空间表达式为

$$\begin{cases} M_1 \ltimes_{i=1}^n x_i = \delta_{\mu_1}^{d_1}, \\ \qquad \cdots\cdots \\ M_s \ltimes_{i=1}^n x_i = \delta_{\mu_s}^{d_s}, \end{cases} \tag{2.3.38}$$

这里, $x_i = \vec{X}_i$, $i = 1, \cdots, n$, M_j 是 φ_j 的结构矩阵, $\delta_{\mu_1}^{d_1} = \vec{c}_j$, $j = 1, \cdots, s$.

类似于布尔网络, 我们有如下结论.

定理 2.3.3　考虑混合值网络 (2.3.35) 及给定分割 $X{-}Y$. 记相应逻辑矩阵为 $T_{\Sigma}^{(X,Y)}$. 又 $F : X \to Y$ 为一逻辑映射, 其结构矩阵为 M_F.

(i) F 是网络 (2.3.35) 的前提解, 当且仅当,

$$M_F \leqslant T_{\Sigma}^{(X,Y)}. \tag{2.3.39}$$

(ii) F 是网络 (2.3.35) 的推论解, 当且仅当,

$$T_{\Sigma}^{(X,Y)} \leqslant M_F. \tag{2.3.40}$$

下面讨论一个例子.

例 2.3.6　考察下述网络

$$\begin{cases} M_1 x_1 x_2 x_3 x_4 = \delta_3^2, \\ M_2 x_1 x_2 x_3 x_4 = \delta_2^1, \end{cases} \tag{2.3.41}$$

这里 $x_1, x_4 \in \Delta_2$, $x_2, x_3 \in \Delta_3$,

$$\begin{aligned} M_1 = \delta_3[&1,1,2,1,3,3,1,2,1,2,1,3,\\ &3,2,1,2,2,2,3,3,2,2,3,2,\\ &3,2,1,3,2,2,2,3,1,2,3,1];\\ M_2 = \delta_2[&1,2,1,1,2,1,2,2,1,1,2,1,\\ &2,2,1,1,2,1,1,2,1,2,1,2,\\ &2,1,1,1,1,2,1,1,2,2,2,1]. \end{aligned}$$

(i) 设 (X, Y) 为

$$X = \{x_1, x_2\}, \quad Y = \{x_3, x_4\},$$

记 $x := x_1x_2$, $y := x_3x_4$.

关于这个分割的真值矩阵见表 2.3.4.

表 2.3.4 (2.3.41) 的真值矩阵

x	y					
	δ_6^1	δ_6^2	δ_6^3	δ_6^4	δ_6^5	δ_6^6
δ_6^1	0	0	0	0	0	1
δ_6^2	0	0	0	0	1	0
δ_6^3	1	0	0	1	0	0
δ_6^4	0	1	1	0	0	0
δ_6^5	0	0	0	0	1	0
δ_6^6	0	0	1	0	0	0

根据定理 2.3.3, 不难验证, 在这个分割下, (2.3.41) 有 4 个前提解, 记作

$$y = T_i x, \quad i = 1, 2, 3, 4,$$

这里,

$$\begin{aligned} T_1 &= \delta_6[3,4,4,3,2,1],\\ T_2 &= \delta_6[3,4,6,3,2,1],\\ T_3 &= \delta_6[3,4,4,3,5,1],\\ T_4 &= \delta_6[3,4,6,3,5,1]. \end{aligned}$$

- $i = 1$: 则

$$\begin{cases} x_3 = M_1^1 x_1 x_2 = \left(I_3 \otimes \mathbf{1}_2^{\mathrm{T}}\right) T_1 x_1 x_2,\\ x_4 = M_1^2 x_1 x_2 = \left(\mathbf{1}_3^{\mathrm{T}} \otimes I_2\right) T_1 x_1 x_2. \end{cases}$$

我们有

$$M_1^1 = \delta_3[2,2,2,2,1,1], \quad M_1^2 = \delta_2[1,2,2,1,2,1].$$

类似地可得

- $i=2$:

$$M_2^1=\delta_3[2,2,3,2,1,1],\quad M_2^2=\delta_2[1,2,2,1,2,1];$$

- $i=3$:

$$M_3^1=\delta_3[2,2,2,2,3,1],\quad M_3^2=\delta_2[1,2,2,1,1,1];$$

- $i=4$:

$$M_4^1=\delta_3[2,2,3,2,3,1],\quad M_4^2=\delta_2[1,2,2,1,1,1].$$

根据定理 2.3.3, 网络 (2.3.35) 关于分割 $X-Y$ 没有推论解.

(ii) 设 (X,Y) 为

$$X=\{x_1,x_2,x_3\},\quad Y=\{x_4\},$$

记 $x:=x_1x_2x_3$, $y:=x_4$.

关于这个分割的真值矩阵

$$T_\Sigma^{(X,Y)}=\begin{bmatrix}0&1&0&0&0&0&0&0&0&0&1&0&0&0&1&1&0&0\\0&0&0&0&1&0&0&1&1&0&0&0&1&0&0&0&0&0\end{bmatrix}.$$

根据定理 2.3.3, 网络 (2.3.35) 关于分割 $X-Y$ 没有前提解. 但它有 2^{10} 个推论解

$$y=M_F^ix_1x_2x_3,\quad i=1,2,\cdots,2^{10}.$$

例如

$$M_F^1=\begin{bmatrix}1&1&1&1&0&1&1&0&0&1&1&1&0&1&1&1&1&1\\0&0&0&0&1&0&0&1&1&0&0&0&1&0&0&0&0&0\end{bmatrix},$$

$$M_F^2=\begin{bmatrix}1&1&1&1&0&1&1&0&0&1&1&1&0&1&1&1&1&0\\0&0&0&0&1&0&0&1&1&0&0&0&1&0&0&0&0&1\end{bmatrix},$$

等等.

2.4　Ledley 解的应用

2.4.1　广义隐函数存在定理

回到布尔网络 (2.3.10), 回忆前面对它前提解和推论解的讨论, 不难推出以下结论.

定理 2.4.1　考察布尔网络 (2.3.10) (或曰 Σ), 对于给定的分割 $X-Y$, 有如下结论:

(i) Σ 在分割 $X-Y$ 下既有前提解又有推论解, 当且仅当, 真值矩阵是一个逻辑矩阵, 即

$$T_{\Sigma}^{(X,Y)} \in \mathcal{L}_{2^q \times 2^p}. \tag{2.4.1}$$

(ii) 如果 (2.4.1) 成立, 则存在唯一的前提解 $F: X \to Y$, 它同时也是唯一的推论解, 并且, 其结构矩阵

$$M_F = T_{\Sigma}^{(X,Y)}. \tag{2.4.2}$$

定义 2.4.1　如果布尔网络 (2.3.10) (或曰 Σ), 对于给定的分割 $X-Y$, 其真值矩阵是一个逻辑矩阵, 则它所唯一确定的函数 $F: X \to Y$, 称为广义隐函数.

注 2.4.1　(i) 广义隐函数与原来的布尔网络是等价的.

(ii) 不难验证, 定义 2.2.1 中给出的隐函数也是广义隐函数. 只是广义隐函数在网络方程个数与隐函数个数等方面不需特别约定. 这在经典连续系统里一般是不会出现的.

(iii) 对于 k 值或混合值网络 (2.3.35), 定理 2.4.1 一样成立. 因此, 可以用这种方法定义混合值逻辑系统的隐函数.

下面这个例子展示了广义隐函数的多样性.

例 2.4.1　考虑下述代数状态空间下的布尔系统:

$$\begin{cases} M_1 x_1 x_2 x_3 x_4 = \delta_2^1, \\ M_2 x_1 x_2 x_3 x_4 = \delta_2^2, \end{cases} \tag{2.4.3}$$

这里, $x_i = \vec{X}_i \in \Delta_2$, $i = 1, 2, 3, 4$,

$$M_1 = \delta_2[1,2,1,2,2,1,2,1,1,2,2,1,1,2,1,1];$$
$$M_2 = \delta_2[2,1,1,1,2,1,1,2,1,1,2,2,2,1,1,1].$$

(i) 给定分割 (X^1, Y^1), 这里 $X^1 = \{X_1, X_2\}$, $Y^1 = \{X_3, X_4\}$. 令 $x = x_1 x_2$, $y = x_3 x_4$, 容易算得

$$T_{(2.4.3)}^{(X^1,Y^1)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}. \tag{2.4.4}$$

根据定理 2.4.1, 存在广义隐函数

$$y = T_{(2.4.3)}^{(X^1,Y^1)} x. \tag{2.4.5}$$

于是有

$$x_3 = (I_2 \otimes \mathbf{1}_2^{\mathrm{T}})y = \delta_2[1,2,2,1]x_1x_2,$$
$$x_4 = (\mathbf{1}_2^{\mathrm{T}} \otimes I_2)y = \delta_2[1,2,2,1]x_1x_2.$$

回到逻辑系统, 有

$$\begin{cases} X_3 = X_1 \leftrightarrow X_2, \\ X_4 = X_1 \leftrightarrow X_2. \end{cases} \tag{2.4.6}$$

(ii) 给定分割 (X^2, Y^2), 这里 $X^2 = \{X_1, X_3\}$, $Y^2 = \{X_2, X_4\}$. 令 $x = x_1x_3$, $y = x_2x_4$, 略去过程, 可得广义隐函数

$$\begin{cases} X_2 = X_1 \leftrightarrow X_3, \\ X_4 = X_3. \end{cases} \tag{2.4.7}$$

(iii) 给定分割 (X^3, Y^3), 这里 $X^3 = \{X_1, X_4\}$, $Y^3 = \{X_2, X_4\}$. 令 $x = x_1x_3$, $y = x_2x_4$, 略去过程, 可得广义隐函数

$$\begin{cases} X_2 = X_1 \leftrightarrow X_4, \\ X_3 = X_4. \end{cases} \tag{2.4.8}$$

(iv) 给定分割 (X^4, Y^4), 这里 $X^4 = \{X_1, X_4\}$, $Y^4 = \{X_2, X_4\}$. 令 $x = x_1x_3$, $y = x_2x_4$, 根据定理 2.4.1, 在这个分割下不存在广义隐函数.

(v) 给定分割 (X^5, Y^5), 这里 $X^5 = \{X_2, X_4\}$, $Y^5 = \{X_1, X_3\}$. 令 $x = x_2x_4$, $y = x_1x_3$, 略去过程, 可得广义隐函数

$$\begin{cases} X_1 = X_2 \leftrightarrow X_4, \\ X_3 = X_4. \end{cases} \tag{2.4.9}$$

(vi) 给定分割 (X^6, Y^6), 这里 $X^6 = \{X_2, X_3\}$, $Y^6 = \{X_1, X_4\}$. 令 $x = x_2x_3$, $y = x_1x_4$, 略去过程, 可得广义隐函数

$$\begin{cases} X_1 = X_2 \leftrightarrow X_3, \\ X_4 = X_3. \end{cases} \tag{2.4.10}$$

(vii) 给定分割 (X^7, Y^7), 设 $|X^7| = 1$, $|Y^7| = 3$ (或 $|X^7| = 3$, $|Y^7| = 1$), 不难发现, 此时相应的真值矩阵都不是逻辑矩阵, 因此, 在这种分割下不存在广义隐函数.

下面这个例子给出混合值逻辑系统的广义隐函数.

例 2.4.2 考虑如下混合值逻辑系统:

$$\varphi(X_1,X_2,X_3):=X_1\Box(X_2\nabla X_3)=0.5, \tag{2.4.11}$$

这里, $X_1\in\mathcal{D}_2$, X_2, $X_3\in\mathcal{D}_3$, $\nabla:\mathcal{D}_3\times\mathcal{D}_3\to\mathcal{D}_3$, $\Box:\mathcal{D}_2\times\mathcal{D}_3\to\mathcal{D}_3$. 它们的结构矩阵分别为

$$M_\nabla=\delta_3[1,2,3,2,3,1,3,1,2];$$
$$M_\Box=\delta_3[3,3,2,1,1,2].$$

于是有

$$\begin{aligned}M_\varphi&=M_\Box\,(I_2\otimes M_\nabla)\\&=\delta_3[3,3,2,3,2,3,2,3,3,1,1,2,1,2,1,2,1,1].\end{aligned}$$

考虑分割 $X=(X_1,X_2)$, $Y=(X_3)$. 不难算得

$$T^{(A,B)}_{(2.4.11)}=\delta_3[3,2,1,3,2,1].$$

这是一个逻辑矩阵, 故有广义隐函数. 易知, 这个隐函数是

$$X_3=\neg X_2.$$

2.4.2 检测问题

检测问题可以看作真值矩阵的一个应用, 具体描述如下: 设有两组变量 X–Y, $X=(X_1,X_2,\cdots,X_p)$, $Y=(Y_1,Y_2,\cdots,Y_q)$. 它们可能有关联, 可能没有关联. 作一个 $q\times p$ 的矩阵 $T=(t_{i,j})\in\mathcal{B}_{q\times p}$, 如果 Y_i 与 X_j 相关, 则令 $t_{i,j}=1$, 否则, 令 $t_{i,j}=0$. T 称为关于 $X-Y$ 的真值矩阵.

这种问题很普遍, 如疾病诊断, 假定某患者的可能疾病有 $d_1,\cdots,d_n$, 化验手段有 $e_1,\cdots,e_m$. 如果化验手段 e_i 对疾病 d_j 有效 (呈阳性), 则令 $t_{i,j}=1$, 否则令 $t_{ij}=0$. 这样, 就可以得到真值矩阵 $T=(t_{ij})\in\mathcal{B}_{m\times n}$. 又如采矿问题, 设 d_j 是矿石可能含有的元素, e_i 为试剂, 检测的目的是测定矿石是否含元素 d_j.

对于检测矩阵, 一个典型问题是 Ledley 问题[9], 即如何由最少的 e_i 确定 d_j. 假如, 要确诊某患者是否患有疾病 d_i, 准确诊断所需要的最少检测数是多少?

下面给出一个算法.

算法 2.4.1 第一步: 将所检测的列 d_i 中的每个零元素 t_{ji} 所在行 j 中的 e_j 改为 $\neg e_j$, 然后将 t_{js} 改为 $\neg t_{js}$, $s=1,\cdots,n$.

第二步: 寻找零最多的行 α, 做 e_α 检验. 然后检查 $t_{\alpha,\beta}$ $(\forall\beta)$. 如果 $t_{\alpha,\beta}=0$ 则将 β 列删去. 最后删除第 α 行.

第三步: 对剩余的矩阵重复第二步, 直到

(i) 情况 1: 只剩检测列 d_i, 则历次所得 e_α 构成有效检测序列.

(ii) 情况 2: 剩余多于一列 (检测列) 但已无带零的行, 则检测无法唯一确定疾病.

注 2.4.2　算法 2.4.1 的合理性在于, 它在每一步出局的疾病数都是最多的, 因此, 它有较少的检验数. 但这仍然不能保证这是最少检验数, 因为不同路径的每一步出局疾病数不能相比.

如果情况 2 发生, 那说明留下的每一列 (每一种病) 都是有可能的, 但已删去的就不可能了.

用一个例子来说明算法 2.4.1.

例 2.4.3 [83]　设真值矩阵 T 如表 2.4.1所示, 找出尽可能少的检测数确定 d_1.

表 2.4.1　检测 (真值) 表-1

	d_1	d_2	d_3	d_4	d_5
e_1	1	0	0	1	0
e_2	0	1	1	0	0
e_3	1	0	0	0	1
e_4	1	1	0	0	1

第一步: 在本例中, 检测的列为 d_1. 该列中有一个零, 为 $t_{2,1}$. 所以将第二行取 “非”. 注意: 在第一步完成后, 第 d_1 列元素全部为 1 (表 2.4.2).

表 2.4.2　检测 (真值) 表-2

	d_1	d_2	d_3	d_4	d_5
e_1	1	0	0	1	0
$\neg e_2$	1	0	0	1	1
e_3	1	0	0	0	1
e_4	1	1	0	0	1

第二步: 取 $\alpha = 1$, (由于第 1 行零个数为 3) 做检验 e_1. 然后删去所有 $t_{1,\beta} = 0$ 的列 β. 即, $\beta = 2, 3, 5$. 再删去第 1 行. 剩下的矩阵见表 2.4.3.

表 2.4.3　检测 (真值) 表-3

	d_1	d_4
$\neg e_2$	1	1
e_3	1	0
e_4	1	0

第三步: 对表 2.4.3 选 $\alpha = 2$ 或 $\alpha = 3$, 即, 做检验 e_3 或 e_4. 然后删去 d_4, 仅余 d_1.

结论: $\{e_1,\ e_3\}$ 或 $\{e_1,\ e_4\}$ 均为有效检验.

当然, 在第二步, 也可取 $\alpha = 3$, (由于第 3 行零个数也为 3) 做检验 e_3. 然后删去所有 $t_{3,\beta} = 0$ 的列 β. 即, $\beta = 2, 3, 4$. 再删去第 3 行. 剩下的矩阵见表 2.4.4.

表 2.4.4　检测 (真值) 表 -4

	d_1	d_5
e_1	1	0
$\neg e_2$	1	1
e_4	1	1

在第三步, 对表 2.4.4 选 $\alpha = 1$, 即, 做检验 e_1. 结论: $\{e_3,\ e_1\}$ 也是有效检验.

下面讨论可检测问题, 即给了一张检测表, 问何时 d_j 是可检测的. 下面这个结论很简单.

命题 2.4.1　给定一个检测真值表 $T = (t_{ij}) \in \mathcal{B}_{m\times n}$.

(i) $d_j, j = 1, \cdots, n$ 是可检测的, 当且仅当

$$\mathrm{Col}_s(T) \neq \mathrm{Col}_j(T), \quad s \neq j.$$

(ii) d_j, $j = 1, \cdots, n$ 可检测, 当且仅当

$$\mathrm{Col}_p(T) \neq \mathrm{Col}_q(T), \quad 1 \leqslant p < q \leqslant n.$$

第 3 章　逻辑动态系统的结构分析

本章所讨论的逻辑动态系统包括很广泛的一类系统，它有如下几个特征: ① 这是一个时间离散的动态系统; ② 所有的动态变量都只能取有限多个值. 因此, 它也被称为离散时间、离散状态的动态系统[146]. 从逻辑角度看, 它可以是经典逻辑 (二值逻辑)、k 值逻辑或混合值逻辑. 当然, 本章所提出的方法也可以应用到一些本质上非逻辑的有限值动态过程, 譬如有限博弈等[27].

布尔网络 (二值逻辑网络) 是这类动态系统中最简单也最有用的一类. 并且, 这类系统的性质容易推广到 k 值逻辑网络或混合值逻辑网络上去. 因此, 布尔网络是我们讨论的重点[22].

3.1　布尔网络的拓扑结构

在 20 世纪 60 年代初, 法国生物学家 Jacob 和微生物学家 Monod 发现细胞中的调节基因可以打开或关闭其他基因, 从而形成基因网络. 他们因此在 1965 年获得了诺贝尔生理学或医学奖[116]. 在他们发现的启发下, 美国学者 Kauffman 提出用布尔网络来刻画细胞与基因调控网络. 此后, 相关研究取得很大成功, 从而使布尔网络成为系统生物学研究的一个有效工具[74].

自 2008 年开始, 矩阵半张量积方法开始被用于描述和分析布尔网络, 并用于其控制设计. 由于矩阵半张量积在逻辑系统建模与控制中的有效性, 它的使用大大推动了布尔网络及其控制理论的发展, 文献 [22] 首次对基于矩阵半张量积的布尔网络控制理论给出一个框架性的总结和系统性的介绍. 此后十年, 布尔网络控制的矩阵半张量积方法得到快速发展, 新的理论与应用成果层出不穷. 国内外学者的若干专著 [2, 5, 27, 93, 146] 与综述文章 [47, 93, 99, 104] 反映了这些进展.

3.1.1　布尔网络的代数表示

一个布尔网络可以用一个网络图来描述. 网络中每个结点在每一个时刻可取一个逻辑值, 即 0 或 1. 在基因调控网络中 1 表示这个基因被激活, 而 0 表示它未被激活. 每一条输入边表示另一个结点对它作用. 而一个结点的演化值 (即它在 $t+1$ 时刻的值) 则由这些作用的一个逻辑函数来确定.

例如, 图 3.1.1 是一个具有四个结点的布尔网络. 结点 X_1, X_2, X_3, X_4 在每个时刻 t 可取不同的逻辑值. 每个结点在 $t+1$ 时刻的值是它的邻域结点在 t 时刻的

值的一个逻辑函数. 网络图通常只画出邻域关系, 这时, 我们将作用于某一结点的那些结点称为该点的邻域. 从图上看, 有边从这些点出发而指向该结点的点即为邻域点, 邻域点个数称为该点的入度. 为了应用的方便, 我们用 $U(X)$ 记包括 X 自身的邻域. 在图 3.1.1 中, X_1 的邻域为 $\{X_1, X_2, X_4\}$, 记作 $U(X_1) = \{X_1, X_2, X_4\}$; X_2 的邻域为 $U(X_2) = \{X_1, X_2, X_3\}$; X_3 的邻域为 $U(X_3) = \{X_3, X_4\}$; X_4 的邻域为 $U(X_4) = \{X_2, X_3, X_4\}$.

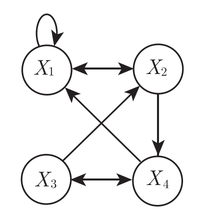


图 3.1.1 布尔网络

仅靠网络图不足以决定一个布尔网络的性质, 因为邻域点如何作用于一个结点并未在图中标出. 因此, 我们还需要将网络的逻辑动态过程用逻辑函数表出. 例如, 设该布尔网络的动态方程可表示为

$$\begin{cases} X_1(t+1) = X_1(t) \wedge (X_2(t) \vee X_4(t)), \\ X_2(t+1) = X_1(t) \leftrightarrow X_3(t), \\ X_3(t+1) = \neg X_4(t), \\ X_4(t+1) = X_2(t) \bar{\vee} X_3(t), \end{cases} \tag{3.1.1}$$

这里, $X_1(t), X_2(t), X_3(t), X_4(t) \in \mathcal{D} = \{0, 1\}$.

一般地说, 一个 n 个结点的布尔网络, 其动力学演化方程可表示为

$$\begin{cases} X_1(t+1) = F_1(X_1(t), \cdots, X_n(t)), \\ X_2(t+1) = F_2(X_1(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), \cdots, X_n(t)), \end{cases} \tag{3.1.2}$$

这里, $X_i(t) \in \mathcal{D}$, $i = 1, 2, \cdots, n$.

下面简单回忆一下演化方程的代数状态空间表示, 详情可参见第一卷的第 5 章与第 6 章.

逻辑变量 $X \in \mathcal{D} = \{0, 1\}$. 它的向量表示记作 $x = \vec{X}$, 定义为

$$x = \vec{X} = \begin{bmatrix} X \\ 1 - X \end{bmatrix} = \begin{cases} \delta_2^1, & X = 1, \\ \delta_2^2, & X = 0. \end{cases} \tag{3.1.3}$$

令 $x_i = \vec{X}_i$, $i = 1, 2, \cdots, n$. 于是, (3.1.2) 可表示为

$$\begin{cases} x_1(t+1) = M_1 x(t), \\ x_2(t+1) = M_2 x(t), \\ \qquad \cdots\cdots \\ x_n(t+1) = M_n x(t), \end{cases} \tag{3.1.4}$$

这里, $M_i \in \mathcal{L}_{2\times 2^n}$ 是 F_i 的结构矩阵, $i = 1, 2, \cdots, n$,

$$x(t) = \ltimes_{i=1}^n x_i(t).$$

式 (3.1.4) 称为网络 (3.1.2) 的分量代数状态空间表示, 这里 “代数” 指演化方程是用矩阵式表示, 而不是用逻辑式表示.

命题 3.1.1 [22]　记 $x(t) = \ltimes_{j=1}^n x_j(t)$, 则 (3.1.2) 可表示为

$$x(t+1) = Mx(t), \tag{3.1.5}$$

这里 M 称为网络 (3.1.2) 的结构矩阵.

定义 3.1.1　设 $A \in \mathcal{M}_{p\times n}$, $B \in \mathcal{M}_{q\times n}$, 则其 Khatri-Rao 积, 记作 $A * B$, 定义为

$$\begin{aligned} A * B := \big[&\mathrm{Col}_1(A) \ltimes \mathrm{Col}_1(B), \mathrm{Col}_2(A) \ltimes \mathrm{Col}_2(B), \cdots, \\ &\mathrm{Col}_n(A) \ltimes \mathrm{Col}_n(B) \big] \in \mathcal{M}_{pq\times n}. \end{aligned} \tag{3.1.6}$$

注 3.1.1　网络 (3.1.2) 的结构矩阵 M 的计算有两种常用方法:

(i) 直接方法. 先求出分量关于邻域的代数状态空间表达式

$$\begin{cases} x_1(t+1) = M_1^U \prod\limits_{i\in U(X_1)} x_i(t), \\ x_2(t+1) = M_2^U \prod\limits_{i\in U(X_2)} x_i(t), \\ \qquad \cdots\cdots \\ x_n(t+1) = M_n^U \prod\limits_{i\in U(X_n)} x_i(t), \end{cases} \tag{3.1.7}$$

然后将方程 (3.1.7) 两边分别直接相乘, 得到

$$x(t+1) = \prod_{j=1}^n \left[M_j^U \prod_{i\in U(X_i)} x_i(t) \right]. \tag{3.1.8}$$

这里 $x(t) := \prod_{i=1}^n x_i(t)$. 最后, 经变量的换序与降阶使其变为标准形式.

(ii) 间接方法. 先求出分量代数状态空间表达式 (3.1.4), 然后用以下公式求解.

$$M = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{2^n \times 2^n}, \tag{3.1.9}$$

这里, $*$ 为 Khatri-Rao 积.

式 (3.1.5) 称为网络 (3.1.2) 的 (整体) 代数状态空间表示.

为了应用方便, 我们将常用逻辑算子及其结构矩阵列成表格 (表 3.1.1).

表 3.1.1 逻辑算子及其结构矩阵

名称	符号	矩阵记号	结构矩阵
非	$\neg$	M_n	$\delta_2[2,1]$
合取	$\wedge$	M_c	$\delta_2[1,2,2,2]$
析取	$\vee$	M_d	$\delta_2[1,1,1,2]$
蕴涵	$\to$	M_i	$\delta_2[1,2,1,1]$
等价	$\leftrightarrow$	M_e	$\delta_2[1,2,2,1]$
异或	$\bar{\vee}$	M_p	$\delta_2[2,1,1,2]$

下面给出一个简单命题, 它对网络演化的代数表示是很重要的.

命题 3.1.2 设 $x_i \in \Delta_{k_i}$, $i = 1,2,\cdots,n$. $x = \ltimes_{i=1}^n x_i$. 记

$$p_i = \prod_{j=1}^{i} k_j;\ q_i = \prod_{j=i+1}^{n} k_j, \quad i = 1,2,\cdots,n,$$

则

$$x_i = \left[\mathbf{1}_{p_{i-1}} \otimes I_{n_i} \otimes \mathbf{1}_{q_i}\right] x. \tag{3.1.10}$$

证明 因为对于列向量集合半张量积与张量积是一样的, 因此

$$x = x^{f_i} \otimes x_i \otimes x^{r_i}, \tag{3.1.11}$$

这里

$$x^{f_i} = \ltimes_{j=1}^{i-1} x_j,$$
$$x^{r_i} = \ltimes_{j=i+1}^{n} x_j.$$

将 (3.1.11) 代入 (3.1.10) 右式, 利用矩阵张量积性质, 即得 (3.1.10) 左式. □

由以上证明可得如下推论.

推论 3.1.1 (i) 设 $x_i \in \Delta_{k_i}$, $i = 1, 2, \cdots, n$, $J = \{j_1, j_2, \cdots, j_s\} \subset \{1, 2, \cdots, n\}$, 则

$$\ltimes_{r=1}^{s} x_{j_r} = (S_1 \otimes S_2 \otimes \cdots \otimes S_n) x, \tag{3.1.12}$$

这里

$$S_i = \begin{cases} I_{k_i}, & i \in J, \\ \mathbf{1}_{k_i}, & i \notin J. \end{cases}$$

(ii) 设 $x_i \in \Upsilon_{k_i}$, $i = 1, 2, \cdots, n$ (即 x_i 为概率向量), 则 (3.1.12) 仍然成立.

例 3.1.1 考察布尔网络 (3.1.1), 容易算出

$$\begin{aligned} x_1(t+1) &= M_c x_1(t) M_d x_2(t) x_4(t) \\ &= M_c (I_2 \otimes M_d) x_1(t) x_2(t) x_4(t) \\ &= M_c (I_2 \otimes M_d)(I_4 \otimes \mathbf{1}_2^{\mathrm{T}} \otimes I_2) x_1(t) x_2(t) x_3(t) x_4(t) \\ &= M_1 x(t), \end{aligned}$$

这里 $x(t) = \ltimes_{i=1}^{4} x_i(t)$,

$$\begin{aligned} M_1 &= M_c (I_2 \otimes M_d)(I_4 \otimes \mathbf{1}_2^{\mathrm{T}} \otimes I_2) \\ &= \delta_2[1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2]. \end{aligned}$$

类似地, 可算得

$$\begin{aligned} M_2 &= M_e (I_2 \otimes \mathbf{1}_4^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}}) \\ &= \delta_2[1,1,2,2,1,1,2,2,2,2,1,1,2,2,1,1]. \\ M_3 &= M_n (\mathbf{1}_8^{\mathrm{T}} \otimes I_2) \\ &= \delta_2[2,1,2,1,2,1,2,1,2,1,2,1,2,1,2,1]. \\ M_4 &= M_p (\mathbf{1}_2^{\mathrm{T}} \otimes I_4 \otimes \mathbf{1}_2^{\mathrm{T}}) \\ &= \delta_2[2,2,1,1,1,1,2,2,2,2,1,1,1,1,2,2]. \end{aligned}$$

最后可得

$$\begin{aligned} M &= M_1 * M_2 * M_3 * M_4 \\ &= \delta_{16}[4,2,7,5,3,1,8,14,16,14,11,9,15,13,12,10]. \end{aligned}$$

定义 3.1.2 $X^* \in \mathcal{D}^n$ 称为布尔网络 (3.1.2) 的不动点 (fixed point), 如果由 $X(t) = X^*$ 可得 $X(t+1) = X^*$. $(X_0^*, X_1^*, \cdots, X_p^* = X_0^*)$ 称为布尔网络 (3.1.2) 的

长度为 p 的极限环 (cycle), 如果 $X_i^* \neq X_j^*$, $0 \leqslant i \neq j \leqslant p-1$, 并且, 由 $X(t)=X_0^*$ 可得 $X(t+i)=X_i^*$, $i=1,\cdots,p$. 不动点和极限环统称为吸引子 (attractor).

注 3.1.2 (1) 一个 n 个结点的布尔网络, 它的状态可以用三种方法表示:

(i) $(X_1,X_2,\cdots,X_n)$;

(ii) $(x_1,x_2,\cdots,x_n)$, 这里 $x_i=\vec{X}_i$, $i=1,2,\cdots,n$;

(iii) $x=\ltimes_{i=1}^n x_i$.

(i) 与 (ii) 的等价性是显见的. 要证明 (iii) 与 (ii) 的等价性, 显然, 从 (ii) 可唯一地确定 (iii), 那么, 只要证明由 (iii) 可唯一地确定 (ii) 即可. 利用 (3.1.10) 可得

$$\left(\mathbf{1}_{2^{i-1}}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-i}}\right) x = x_i, \quad i=1,2,\cdots,n. \tag{3.1.13}$$

(2) 由上述等价性可知, (3.1.2) 的不动点和极限环在向量形式下即为 (3.1.5) 的不动点和极限环. 因此, 要计算吸引子只要在代数形式下计算即可.

3.1.2 吸引子的计算

由于一个布尔网络只有有限个结点, 而且, 每个结点都只能取有限个值, 因此, 一个布尔网络的每一条轨线都要收敛到一个吸引子上去. 所以, 不动点与极限环是一个布尔网络最主要的拓扑特征. 寻找一个布尔网络的所有不动点与极限环也成为布尔网络理论研究的一个经典问题. 在逻辑表达式下直接讨论这个问题是困难的, 因此, 经典方法多半只能具体网络具体分析. 于是, 早期的文献通常是一篇文章讨论一个或两个特殊系统, 难以找到一般规律[42, 54, 60, 63].

下面我们以矩阵半张量积为工具, 通过布尔网络的代数状态空间表示, 给出一般不动点与极限环的计算公式.

先考虑不动点如下.

定理 3.1.1 考察布尔网络 (3.1.2), 其代数状态空间表示为 (3.1.5). (3.1.2) 的不动点个数 N_e 为

$$N_e = \mathrm{tr}(M). \tag{3.1.14}$$

证明 设 $\delta_{2^n}^i$ 为其一个不动点. 注意到 $M\delta_{2^n}^i = \mathrm{Col}_i(M)$. 显然 $\delta_{2^n}^i$ 为其不动点, 当且仅当, $\mathrm{Col}_i(M)=\delta_{2^n}^i$. 设 $m_{ii}=1$, 则称 $\mathrm{Col}_i(M)$ 为对角非零列. 显然, 对角非零列即不动点. 于是 (3.1.14) 显见. □

下面考虑布尔网络 (3.1.2) 的极限环. 先介绍一个记号: 设 $k\in\mathbb{Z}_+$ 为一正整数, $s\in\mathbb{Z}_+$ 称为 k 的恰当因子, 如果 $s<k$ 并且 $k/s\in\mathbb{Z}_+$ (即 $s|k$). k 的真因子集合记作 $\mathcal{P}(k)$. 假如, $\mathcal{P}(8)=\{1,2,4\}$, $\mathcal{P}(12)=\{1,2,3,4,6\}$, 等等.

利用与不动点类似的讨论, 可以找到极限环个数的计算公式.

定理 3.1.2　记长度为 s 的极限环个数为 N_s, 则 N_s 可由下列递推公式算得

$$\begin{cases} N_1 = N_e, \\ N_s = \dfrac{\operatorname{tr}(M^s) - \sum\limits_{k\in\mathcal{P}(s)} kN_k}{s}, \quad 2 \leqslant s \leqslant 2^n. \end{cases} \tag{3.1.15}$$

证明　首先, 如果 $\delta_{2^n}^i$ 是长度为 s 的极限环上的一个元素, 则 $M^s\delta_{2^n}^i = \delta_{2^n}^i$. 于是, $\operatorname{Col}_i(M^s) = \delta_{2^n}^i$ 为 M^s 的对角非零列. 因此, 环上每个元素都会为 $\operatorname{tr}(M^s)$ 增加一个 1. 注意到, 如果 $\delta_{2^n}^i$ 是长度为 $k < s$ 的环上的一个元素, 并且, $k \in \mathcal{P}(s)$, 我们也会有 $M^s\delta_{2^n}^i = \delta_{2^n}^i$, $\operatorname{Col}_i(M^s)$ 也会对 $\operatorname{tr}(M^s)$ 增加一个 1. 当考虑长度为 s 的环时, 这样的对角非零列应当减去. 至于求和的上限, 显然不可能超过 2^n. 仔细考虑以上情况, 就不难得出公式 (3.1.15). □

注意, 结构矩阵中每一个对角非零列就是一个不动点, 因此, 寻找不动点是一件简单的事情. 下面讨论如何寻找极限环. 对于某个 $1 < s \leqslant 2^n$, 如果

$$\operatorname{tr}(M^s) - \sum_{k\in\mathcal{P}(s)} kN_k > 0, \tag{3.1.16}$$

则 “s” 称为非平凡幂.

设 s 是一个非平凡幂. 记 m_{ii}^s 为 M^s 的第 (i,i) 元素, 记

$$C_s = \{i \mid m_{ii}^s = 1\}, \quad s = 1, 2, \cdots, 2^n$$

且

$$D_s = C_s \bigcap_{i\in\mathcal{P}(s)} C_i^c,$$

这里, C_i^c 是 C_i 的余集.

由前面的构造方法可得到如下结果.

命题 3.1.3　设 $x_0 = \delta_{2^n}^i$, 那么, $\{x_0, Mx_0, \cdots, M^s x_0\}$ 为一长度为 s 的环, 当且仅当, $i \in D_s$.

定理 3.1.2 与命题 3.1.3 给出了一个构造环的简单方法, 下面用一个例子说明构造步骤.

例 3.1.2　考察系统

$$\begin{cases} X_1(t+1) = X_2(t) \wedge X_3(t), \\ X_2(t+1) = \neg X_1(t), \\ X_3(t+1) = X_2(t) \vee X_3(t). \end{cases} \tag{3.1.17}$$

记 $x_i=\vec{X}_i$ 为 X_i 的向量表达, $i=1,2,3$, 则 (3.1.17) 的代数形式为

$$\begin{cases}x_1(t+1)=M_cx_2(t)x_3(t),\\ x_2(t+1)=M_nx_1(t),\\ x_3(t+1)=M_dx_2(t)x_3(t).\end{cases} \tag{3.1.18}$$

令 $x(t)=x_1(t)x_2(t)x_3(t)$, 则可计算 M 如下:

$$\begin{aligned}x(t+1)&=M_cx_2(t)x_3(t)M_nx_1(t)M_dx_2(t)x_3(t)\\ &=M_c(I_4\otimes M_n)x_2(t)x_3(t)x_1(t)M_dx_2(t)x_3(t)\\ &=M_c(I_4\otimes M_n)(I_8\otimes M_d)_2(t)x_3(t)x_1(t)_2(t)x_3(t)\\ &=M_c(I_4\otimes M_n)(I_8\otimes M_d)W_{[2,4]}x_1(t)x_2(t)x_3(t)_2(t)x_3(t)\\ &=M_c(I_4\otimes M_n)(I_8\otimes M_d)W_{[2,4]}x_1(t)x_2(t)W_{[2]}x_2(t)x_3(t)x_3(t)\\ &=M_c(I_4\otimes M_n)(I_8\otimes M_d)W_{[2,4]}(I_4\otimes W_{[2]})x_1(t)\,\mathrm{PR}_2\,x_2(t)\,\mathrm{PR}_2\,x_3(t)\\ &=M_c(I_4\otimes M_n)(I_8\otimes M_d)W_{[2,4]}(I_4\otimes W_{[2]})(I_2\otimes \mathrm{PR}_2)(I_4\otimes \mathrm{PR}_2)x(t).\end{aligned} \tag{3.1.19}$$

于是可得 (3.1.17) 的代数状态空间表示

$$x(t+1)=Mx(t),$$

这里, 网络的结构矩阵为

$$\begin{aligned}M&=M_c(I_4\otimes M_n)(I_8\otimes M_d)W_{[2,4]}(I_4\otimes W_{[2]})(I_2\otimes \mathrm{PR}_2)(I_4\otimes \mathrm{PR}_2)\\ &=\delta_8[3,7,7,8,1,5,5,6].\end{aligned}$$

下面考虑它的吸引子. 不难算得

$$\mathrm{tr}(M^t)=0,\quad t\leqslant 3,$$

并且

$$\mathrm{tr}(M^t)=4,\quad t\geqslant 4.$$

根据定理 3.1.2 可知, (3.1.17) 只有一个长度为 4 的极限环.

观察

$$M^4=\delta_8[1,3,3,1,5,7,7,3].$$

每一个对角非零列均可生成极限环. 例如, 取 $Z=\delta_8^1$, 则可得

$$MZ=\delta_8^3,\quad M^2Z=\delta_8^7,\quad M^3Z=\delta_8^5,\quad M^4Z=Z.$$

将 $x_1(t)$, $x_2(t)$, $x_3(t)$ 返回到逻辑形式, 则不难得到极限环为

$$(1,1,1) \to (1,0,1) \to (0,0,1) \to (0,1,1) \to (1,1,1).$$

称所有吸引子集合为极限集, 记作 Ω, 则所有状态都进入极限集的最小步数称为过渡周期, 记作 T_t.

下面考虑过渡周期: 注意到 M 至多有 $(2^n)^{2^n}$ 个可能值, 定义

$$r := (2^n)^{2^n} + 1, \tag{3.1.20}$$

如果我们构造一个 r 矩阵序列

$$M, M^2, \cdots, M^r,$$

则必有两个相等的矩阵. 令 $r_0 < r$ 为最小的 i 使 M^i 在序列中重现. 于是, 存在一个 $k > i$ 使得 $M^i = M^k$. 准确地说,

$$r_0 = \operatorname*{argmin}_{1 \leqslant i < r} \left\{ M^i \in \{M^{i+1}, M^{i+2}, \cdots, M^r\} \right\}. \tag{3.1.21}$$

显然, 这样的 r_0 存在, 并且有如下结论.

命题 3.1.4 设 r_0 由 (3.1.21) 定义, 那么, 从任意初始状态出发, 轨线经 r_0 步必定进入极限集 Ω.

对于一个给定状态 x_0, 它进入极限集的时间称为它的过渡周期, 记作 $T_t(x_0)$. 显见

$$T_t = \max_{\forall x \in \Delta_{2^n}} \left(T_t(x) \right).$$

实际上, 不难证明 (3.1.21) 给出的就是过渡周期.

定理 3.1.3 (3.1.21) 中的 r_0 就是系统的过渡周期. 即

$$T_t = r_0. \tag{3.1.22}$$

证明 首先, 设

$$M^{r_0} = M^{r_0+T}, \tag{3.1.23}$$

这里 $T > 0$ 是满足 (3.1.23) 的最小正数. 根据定义, $r_0 + T \leqslant r$. 首先证明: 如果系统有长度为 t 的环, 则 t 是 T 的因子. 我们用反证法: 设 $T (\mathrm{mod}\ t) = s$ 且 $1 \leqslant s < t$. 如果 x_0 是环上的一个点, 则 $M^{r_0} x_0$ 也是这个环上的一个点. 因此,

$$M^{r_0} x_0 = M^{r_0+T} x_0 = M^T M^{r_0} x_0 = M^s (M^{r_0} x_0) \neq M^{r_0} x_0,$$

这导致矛盾.

由 (3.1.23) 和 T_t 的定义可知 $T_t \leqslant r_0$. 要证明 $T_t = r_0$, 我们假定 $T_t < r_0$. 根据定义, 对任何 x, $M^{T_t}x$ 都在环上, 而这个环的长度是 T 的因子. 因此,

$$M^{T_t}x = M^T M^{T_t}x = M^{T_t+T}x, \quad \forall x. \tag{3.1.24}$$

容易证明, 如果对任何 $x \in \Delta_{2^n}$, (3.1.24) 均成立, 那么, $L^{T_t} = L^{T_t+T}$, 这与 r_0 的定义矛盾. □

注 3.1.3 (i) 根据定理 3.1.3, 显然 $r_0 \leqslant 2^n$, 因为过渡周期不可能大于 2^n.

(ii) 设 $r_0 = T_t$ 定义如上, $T > 0$ 为满足 (3.1.23) 的最小正数, 那么, 容易证明: T 是所有环的长度的最小公倍数.

(iii) 从前面的讨论可以看出, 要找出 r_0 我们必须检验 M^s, $s \leqslant r_0 + T$. 也就是, 对

$$s \leqslant 2^{n+1}$$

进行检验.

最后, 我们讨论一下计算复杂性的问题. 回忆式 (3.1.20), 当网络只有 3 个结点, 即 $n = 3$ 时, $r = (2^3)^{2^3} + 1 = 16777217$, 当 $n > 3$ 时, 这更是一个天文数字, 因此, 如果真需要这么长的 $\{M^i \mid i = 0, 1, \cdots\}$ 序列, 这种算法在实际上是不可行的. 实际计算表明, 其实, 出现重复的长度远小于这个数. 因此, 给出一个合理的估计是有意义的. 为了以下的讨论对 k 值和混合值逻辑网络也适用, 我们讨论一个一般的逻辑系统

$$x(t+1) = Mx(t), \quad x(t) \in \Delta_k. \tag{3.1.25}$$

引理 3.1.1 设网络 (3.1.25) 有 s 个极限环 (将不动点视为长度为 1 的极限环), 其长度分别为 ℓ_i, $i = 1, 2, \cdots, s$. 网络的过渡周期为 T_t. 则

$$r \leqslant T_t + \ell, \tag{3.1.26}$$

这里, $\ell = \mathrm{lcm}(\ell_1, \ell_2, \cdots, \ell_s)$ 是极限环长度 $\ell_i = |C_i|$, $i = 1, 2, \cdots, s$ 的最小公倍数.

证明 设网络 (3.1.25) 的状态转移图如图 3.1.2 所示. 这里 C_i 的吸引域为 D_i, $i = 1, 2, \cdots, s$. $T_i = D_i \backslash C_i$ 为 C_i 的过渡点. 从任意一点 $x_0 \in \Delta_k$ 出发, 考虑其轨线: 我们有

$$x_0 \to Mx_0 \to \cdots \to M^{T_t}x_0 \to \cdots \to M^{T_t+\ell}x_0 \to \cdots.$$

根据过渡周期的定义, $M^{T_t}x_0$ 必定在某个 C_j 上. 于是

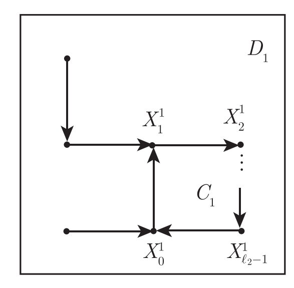
$$M^{T_t}x_0 \to \cdots \to M^{T_t+\ell-1}x_0$$

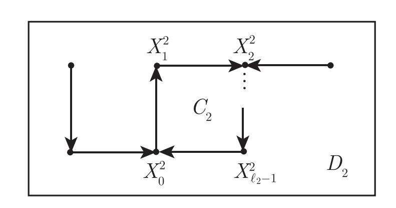
在极限环 C_j 上绕了 ℓ/ℓ_j 圈. 由此可知

$$M^{T_t}x_0 = M^{T_t+\ell}x_0. \tag{3.1.27}$$

因为 (3.1.27) 对任意 $x_0 \in D_k$ 均成立, 所以

$$M^{T_t} = M^{T_t+\ell}. \qquad \square$$





…

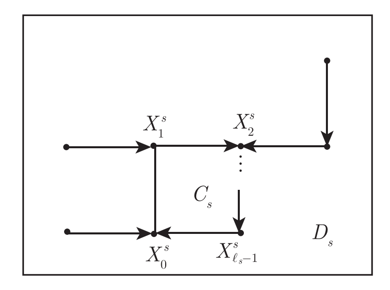


图 3.1.2 状态转移图

命题 3.1.5

$$r \leqslant \lfloor e^{k/e} \rfloor, \tag{3.1.28}$$

这里 $\lfloor z \rfloor$ 是 z 的最大整数下界.

证明 回忆状态转移图 (图 3.1.2). 这里, D_i 是 C_i 的吸引域, $T_i = D_i \backslash C_i$ 是 C_i 的过渡点集. 记 $|T_i| = t_i$, $|D_i| = d_i$. 利用引理 3.1.1 可知

$$r \leqslant T_t + \ell \leqslant T_t + \prod_{i=1}^{s} \ell_i. \tag{3.1.29}$$

设 $t_1 > 0$, 我们可将 T_1 中的一个点移到 C_i 上, 这时, (3.1.29) 的右式只可能增大, 不可能减少, 即

$$T_t + \ell \leqslant T_t + \prod_{i=1}^{s} \ell_i \leqslant (T_t - 1) + (\ell_1 + 1) \prod_{i=2}^{s} \ell_i.$$

同样道理, 可将 T_i 的点全移到 C_i 上, 这时 (3.1.29) 的右式只可能增大, 不可能减少. 于是可知

$$r \leqslant \prod_{i=1}^{s} d_i. \tag{3.1.30}$$

注意到

$$\sum_{i=1}^{s} d_i = k.$$

熟知, 正数的几何平均不大于算术平均, 故

$$r \leqslant \left(\frac{k}{s}\right)^{s}. \tag{3.1.31}$$

下面考虑 s 为何值时 (3.1.31) 右式最大. 不妨取其对数, 即

$$y(s) := s[\ln(k) - \ln(s)].$$
$$y'(s) = \ln(k) - \ln(s) - 1 := 0.$$

则 $s^* = k/e$. 易知, 这是极大点. 注意到 r 是一正整数, 将 s^* 代入 (3.1.31) 即得 (3.1.28). □

下面给出一个例子.

例 3.1.3 考虑三结点布尔网络, 则 $k = 2^3 = 8$. M 有 16777216 种可能情况. 对所有 M 进行穷举计算可知, 当

$$M = \delta_8[2, 3, 1, 5, 6, 7, 8, 4]$$

时相应的 r 最大, 为 $r = 16$.

用 (3.1.28) 估计,

$$r = \lfloor e^{8/e} \rfloor = 18.$$

可见 (3.1.28) 给出的上界估计值还是不错的, 它远小于 16777216.

3.1.3 吸引域

布尔网络的状态集合可以分为两类: ① 吸引子上的点 $\chi \in \Omega$, 它们永远属于固定的吸引子; ② 吸引子外的点 $\chi \in \mathcal{D}^n \backslash \Omega$, 每个点一定会收敛到某个吸引子. 收敛到某个吸引子的点的集合称为这个吸引子的吸引域.

记

$$\Omega := \bigcup_{i=1}^{k} C_i,$$

这里 $\{C_i \mid i = 1, \cdots, k\}$ 为吸引子. 每个吸引子都有自己的吸引域:

定义 3.1.3 (i) $S_i \subset \mathcal{D}^n$ 称为吸引子 C_i 的吸引域, 如果从 S_i 出发的轨线都会收敛到 C_i, 同时, 收敛到 C_i 的初始点都在 S_i 上.

(ii) ξ 称为 η 的前点, 如果, $\eta = \chi(1, \xi)$, 这里 $\chi(t, \chi_0)$ 是初值为 χ_0 的系统轨线.

注 3.1.4 如前所述, 讨论不动点与极限环等可通过结点的向量形式及网络的代数状态空间表达式来讨论, 以下的讨论即在向量与代数式下进行的.

(i) 设 $C \subset \Delta_{2^n}$. 令

$$M^{-1}(C) := \{q \mid Mq \in C\}.$$

那么, 对于单点 p, 其前点为 $M^{-1}(p)$.

(ii) $\Delta_{2^n} = \bigcup_{i=1}^{k} S_i$. 并且, 因为 $\{S_i \mid i = 1, \cdots, k\}$ 是互不相交的, 它们构成状态空间 $\mathcal{X} = \Delta_{2^n}$ 的一个分割.

(iii) 在这个分割里, 每一个 S_i 包括一个吸引子 C_i, 因此, S_i 是 C_i 的吸引域.

最后我们讨论, 如何计算 S_i. 从每一点 $p \in C_i$ 出发, 如果我们能找到 $M^{-1}(p)$, 然后再从 $p_1 \in M^{-1}(p)$ 出发, 找到 $L^{-1}(p_1)$, 一直到至多 $T_t(p)$ 次, 使得该点的前身为空集, 这样就可以找到 S_i. 于是我们有如下的结论.

命题 3.1.6

$$S_i = C_i \cup M^{-1}(C_i) \cup M^{-2}(C_i) \cup \cdots \cup M^{-T_t}(C_i). \tag{3.1.32}$$

于是, 问题的关键变为: 给定 p 如何找到 $M^{-1}(p)$. 下面的结论直接来自定义.

命题 3.1.7

$$\begin{cases} M^{-1}(p) = \left\{ \delta_{2^n}^j \,\middle|\, \mathrm{Col}_j(M) = p \right\}, \\ M^{-k}(p) = \left\{ \delta_{2^n}^j \,\middle|\, \mathrm{Col}_j(M^k) = p \right\}, \quad k = 2, \cdots, T_t. \end{cases} \tag{3.1.33}$$

例 3.1.4 考察系统 (3.1.17) 容易算得 $r_0 = 3$ 并且

$$M^3 = M^7 = \delta_8[5,1,1,5,7,3,3,1].$$

因此, 我们知道, 过渡周期为 $T_t = 3$. 根据命题 3.1.6 及命题 3.1.7, 我们可以从吸引子 C 出发, 从中任选一个点 $p \in C$, 找出 $M^{-1}(p)$, $M^{-2}(p)$ 及 $M^{-3}(p)$.

例如, 选 $p = (0,1,1) \sim \delta_8^5$. 那么, 容易发现, M 中有两个列, $\mathrm{Col}_6(M)$ 及 $\mathrm{Col}_7(M)$, 它们等于 p. 于是, $\delta_8^6 \sim (0,1,0)$ 及 $\delta_8^7 \sim (0,0,1)$ 构成 $M^{-1}(p)$. 但因 $(0,0,1)$ 已在环上, 我们只对 $p_1 = \delta_8^6 \sim (0,1,0)$ 感兴趣. 又因为 $\mathrm{Col}_8(M) = p_1$, 可得 $M^{-1}(p_1) = \{\delta_8^8\}$. 令 $p_2 = \delta_8^8 \sim (0,0,0)$. 只有 $\mathrm{Col}_4(M) = p_2$, 于是有 $p_3 := \delta_8^4 \sim (1,0,0) \in M^{-1}(p_2)$. 于是我们有一条路径 $p_3 \to p_2 \to p_1 \to p$. 再选 $q = (0,0,1) \sim \delta_8^7$. 那么, $\mathrm{Col}_2(M) = \mathrm{Col}_3(M) = q$. 因为 $\delta_8^3 \sim (1,0,1)$ 是个环, 我们选 $q_1 = \delta_8^2 \sim (1,1,0)$. 容易验证 $M^{-1}(q_1) = \varnothing$, 所以 q_1 没有前点. 最后, 我们得到系统 (3.1.17) 的状态转移图如图 3.1.3 所示.

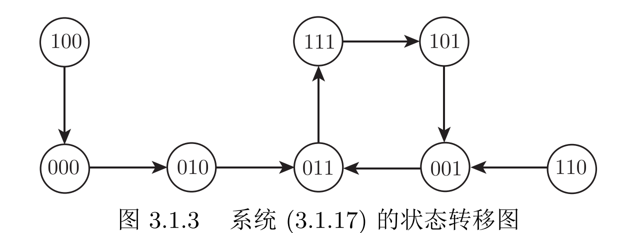


图 3.1.3 系统 (3.1.17) 的状态转移图

3.1.4 布尔网络的例子

本小节回顾相关论文中曾经讨论过的一些布尔网络, 它们或因其结构有代表性, 或因其有生物学背景而引起作者的注意. 我们用一般性方法讨论其拓扑结构, 读者不妨将本节结论与原作者结论做一对比.

例 3.1.5 [135] 考察下列布尔网络

$$\begin{cases} X_1(t+1) = X_1(t) \vee X_2(t), \\ X_2(t+1) = X_1(t) \wedge X_2(t). \end{cases} \tag{3.1.34}$$

记 $x_1 = \vec{X}_1$, $x_2 = \vec{X}_2$, $x(t) = x_1(t)x_2(t)$, 则有

$$\begin{aligned} x(t+1) &= M_d x_1(t) x_2(t) M_c x_1(t) x_2(t) \\ &= M_d (I_4 \otimes M_c) x_1(t) W_{[2]} x_1(t) x_2^2(t) \\ &= M_d (I_4 \otimes M_c)(I_2 \otimes W_{[2]})\, \mathrm{PR}_2\, x_1(t)\, \mathrm{PR}_2\, x_2(t) \\ &= M_d (I_4 \otimes M_c)(I_2 \otimes W_{[2]})\, \mathrm{PR}_2 (I_2 \otimes \mathrm{PR}_2) x(t) \\ &:= M x(t), \end{aligned}$$

这里

$$
\begin{aligned}
M &= M_d(I_4 \otimes M_c)(I_2 \otimes W_{[2]})\,\mathrm{PR}_2(I_2 \otimes \mathrm{PR}_2) \\
&= \delta_4[1,2,2,4].
\end{aligned}
$$

容易算得 $\mathrm{tr}(M) = 3$, 因此, 系统 (3.1.34) 有三个不动点: $\delta_4^1 \sim (1,1)$; $\delta_4^2 \sim (1,0)$, 以及 $\delta_4^4 \sim (0,0)$. 因为 $M^2 = M$, 则 $r_0 = 1$. 最后, 考察 $x = \delta_4^3 \sim (0,1)$, $Mx = \delta_4^2 \sim (1,0)$. 于是可知网络 (3.1.34) 有三个部分: 两个孤立的不动点 $(1,1)$ 和 $(0,0)$, 第三个不动点 $(1,0)$ 有一个吸引域 $S_3 : \{(0,1) \to (1,0)\}$.

例 3.1.6 [42] 考察下列布尔网络

$$
\begin{cases}
X_1(t+1) = X_2(t) \wedge X_3(t), \\
X_2(t+1) = 1 \bar{\vee} X_1(t), \\
X_3(t+1) = X_2(t).
\end{cases}
\tag{3.1.35}
$$

记 $x_i = \vec{X}_i$, $i = 1,2,3$, 容易算出

$$
\begin{aligned}
x(t+1) &= M_c x_2(t) x_3(t) M_p \delta_2^1 x_1(t) x_2(t) \\
&= M_c(I_4 \otimes M_n) x_2(t) x_3(t) x_1(t) x_2(t) \\
&= M_c(I_4 \otimes M_n) W_{[2,4]} x_1(t) x_2(t) x_3(t) x_2(t) \\
&= M_c(I_4 \otimes M_n) W_{[2,4]} x_1(t) x_2(t) W_{[2]} x_2(t) x_3(t) \\
&= M_c(I_4 \otimes M_n) W_{[2,4]} (I_4 \otimes W_{[2]}) x_1(t)\,\mathrm{PR}_2\, x_2(t) x_3(t) \\
&= M_c(I_4 \otimes M_n) W_{[2,4]} (I_4 \otimes W_{[2]})(I_2 \otimes \mathrm{PR}_2) x(t) \\
&:= Mx(t),
\end{aligned}
$$

这里 M 为

$$
M = \delta_8[3,7,8,8,1,5,6,6].
$$

因此

$$
\mathrm{tr}(M^k) = 0, \quad k = 1,2,3,4,
$$

以及

$$
\begin{gathered}
M^5 = \delta_8[1,\ 3,\ 3,\ 3,\ 5,\ 6,\ 8,\ 8], \\
\mathrm{tr}(M^5) = 5.
\end{gathered}
$$

选择 M^5 中任一对角非零列, 例如, 取 $x = \delta_8^1 \sim (1,1,1)$, 则可得长度为 5 的极限环 $x \to Mx \to M^2x \to M^3x \to M^4x \to M^5x = x$, 这里 $Mx = \delta_8^3 \sim (1,0,1)$,

$M^2x = \delta_8^8 \sim (0,0,0)$, $M^3x = \delta_8^6 \sim (0,1,0)$, $M^4x = \delta_8^5 \sim (0,1,1)$, $M^5x = \delta_8^1 \sim (1,1,1)$.

不难算出 $r_0 = 2$ 及 $M^2 = M^7$, 即 $T_t = 2$. 由于 $T = 5$, 不会有长度大于 5 的极限环. 取 $z = \delta_8^2 \sim (1,1,0)$, 那么

$$Mz = \delta_8^7 \sim (0,0,1), \quad M^2z = \delta_8^6 = M^3z.$$

取 $y = \delta_8^4 \sim (1,0,0)$, 那么

$$My = \delta_8^8 = M^2y.$$

图 3.1.4 刻画了这个网络的状态转移过程, 它与 [42] 中的结论一致.

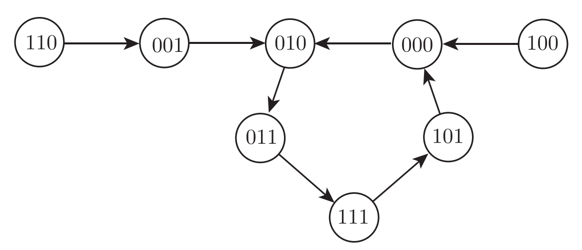


图 3.1.4 例 3.1.6 的状态转移图

下面这个布尔网络是关于细胞生长、分裂和死亡的模型, 它最先在文献 [63] 中提出, 后来, 又在文献 [42] 等中讨论过.

例 3.1.7

$$\begin{cases} X_1(t+1) = X_{11}(t)\bar{\vee}\left(X_{11}(t) \wedge X_8(t)\right), \\ X_2(t+1) = X_1(t)\bar{\vee}\left(X_1(t) \wedge X_3(t)\right), \\ X_3(t+1) = 1\bar{\vee}X_4(t)\bar{\vee}\left(X_4(t) \wedge X_9(t)\right), \\ X_4(t+1) = X_{10}(t) \wedge X_{11}(t), \\ X_5(t+1) = 1\bar{\vee}X_3(t)\bar{\vee}\left(X_3(t) \wedge X_6(t)\right), \\ X_6(t+1) = X_5(t)\bar{\vee}\left(X_5(t) \wedge X_7(t)\right), \\ X_7(t+1) = 1\bar{\vee}\left(X_2(t) \wedge X_5(t)\right), \\ X_8(t+1) = X_6(t)\bar{\vee}\left(X_6(t) \wedge X_7(t)\right), \\ X_9(t+1) = X_8(t)\bar{\vee}\left(X_8(t) \wedge X_9(t)\right), \\ X_{10}(t+1) = X_{10}(t), \\ X_{11}(t+1) = X_{11}(t). \end{cases} \tag{3.1.36}$$

注意到

$$X\bar{\vee}(X \wedge Y) = \neg(X \to Y).$$

利用这个公式可将系统 (3.1.36) 简化, 再将其表示为分量代数状态空间形式 (记 $x_i = \vec{X}_i$, $i = 1, 2, \cdots, 11$)

$$\begin{cases} x_1(t+1) = M_n M_i x_{11}(t) x_8(t), \\ x_2(t+1) = M_n M_i x_1(t) x_3(t), \\ x_3(t+1) = M_i x_4(t) x_9(t), \\ x_4(t+1) = M_c x_{10}(t) x_{11}(t), \\ x_5(t+1) = M_i x_3(t) x_6(t), \\ x_6(t+1) = M_n M_i x_5(t) x_7(t), \\ x_7(t+1) = M_n M_c x_2(t) x_5(t), \\ x_8(t+1) = M_n M_i x_6(t) x_7(t), \\ x_9(t+1) = M_n M_i x_8(t) x_9(t), \\ x_{10}(t+1) = x_{10}(t), \\ x_{11}(t+1) = x_{11}(t). \end{cases} \tag{3.1.37}$$

系统的结构矩阵 M 可计算如下:

$$\begin{aligned} M =& M_n M_i (I_2 \otimes (I_2 \otimes M_n M_i (I_2 \otimes (I_2 \otimes M_i (I_2 \otimes (I_2 \otimes M_c (I_2 \otimes (I_2 \\ & \otimes M_i (I_2 \otimes (I_2 \otimes M_n M_i (I_2 \otimes (I_2 \otimes M_n M_c (I_2 \otimes (I_2 \otimes M_n M_i (I_2 \otimes (I_2 \\ & \otimes M_n M_i)))))))))))))))))(I_2 \otimes W_{[2]}) W_{[2]} (I_{2048} \otimes W_{[2]})(I_{1024} \otimes W_{[2]}) \\ & \ltimes (I_{512} \otimes W_{[2]})(I_{256} \otimes W_{[2]})(I_{128} \otimes W_{[2]})(I_{64} \otimes W_{[2]})(I_{32} \otimes W_{[2]}) \\ & \ltimes (I_{16} \otimes W_{[2]})(I_8 \otimes W_{[2]})(I_4 \otimes W_{[2]})(I_2 \otimes W_{[2]})(I_8 \otimes W_{[2]})(I_4 \otimes W_{[2]}) \\ & \ltimes (I_{256} \otimes W_{[2]})(I_{128} \otimes W_{[2]})(I_{64} \otimes W_{[2]})(I_{32} \otimes W_{[2]})(I_{16} \otimes W_{[2]})(I_8 \otimes W_{[2]}) \\ & \ltimes (I_{32} \otimes W_{[2]})(I_{16} \otimes W_{[2]})(I_{1024} \otimes W_{[2]})(I_{512} \otimes W_{[2]})(I_{256} \otimes W_{[2]}) \\ & \ltimes (I_{128} \otimes W_{[2]})(I_{64} \otimes W_{[2]})(I_{32} \otimes W_{[2]})(I_{4096} \otimes W_{[2]})(I_{2048} \otimes W_{[2]}) \\ & \ltimes (I_{1024} \otimes W_{[2]})(I_{512} \otimes W_{[2]})(I_{256} \otimes W_{[2]})(I_{128} \otimes W_{[2]})(I_{64} \otimes W_{[2]}) \\ & \ltimes (I_{2048} \otimes W_{[2]})(I_{1024} \otimes W_{[2]})(I_{512} \otimes W_{[2]})(I_{256} \otimes W_{[2]})(I_{128} \otimes W_{[2]}) \\ & \ltimes (I_{8192} \otimes W_{[2]})(I_{4096} \otimes W_{[2]})(I_{2048} \otimes W_{[2]})(I_{1024} \otimes W_{[2]})(I_{512} \otimes W_{[2]}) \\ & \ltimes (I_{256} \otimes W_{[2]})(I_{8192} \otimes W_{[2]})(I_{4096} \otimes W_{[2]})(I_{2048} \otimes W_{[2]})(I_{1024} \otimes W_{[2]}) \end{aligned}$$

$$
\begin{aligned}
&\ltimes (I_{512} \otimes W_{[2]})(I_{16384} \otimes W_{[2]})(I_{8192} \otimes W_{[2]})(I_{4096} \otimes W_{[2]})(I_{2048} \otimes W_{[2]}) \\
&\ltimes (I_{1024} \otimes W_{[2]})(I_{2048} \otimes W_{[2]})(I_{32768} \otimes W_{[2]})(I_{16384} \otimes W_{[2]})(I_{8192} \otimes W_{[2]}) \\
&\ltimes (I_{4096} \otimes W_{[2]})(I_{8192} \otimes W_{[2]})(I_{65536} \otimes W_{[2]})(I_{32768} \otimes W_{[2]})(I_{16384} \otimes W_{[2]}) \\
&\ltimes (I_{32768} \otimes W_{[2]})(I_{131072} \otimes W_{[2]})(I_{65536} \otimes W_{[2]})(I_2 \otimes (I_2 \otimes \mathrm{PR}_2(I_2 \otimes (I_2 \\
&\otimes \mathrm{PR}_2(I_2 \otimes \mathrm{PR}_2(I_2 \otimes \mathrm{PR}_2(I_2 \otimes \mathrm{PR}_2(I_2 \otimes \mathrm{PR}_2(I_2 \\
&\otimes \mathrm{PR}_2(I_2 \otimes \mathrm{PR}_2\,\mathrm{PR}_2)))))))))).
\end{aligned}
$$

这是一个 $2^{11} \times 2^{11}$ 矩阵, 这里略去. 可计算相关迹如下:

$$
\mathrm{tr}(M) = 3; \quad \mathrm{tr}(M^9) = 12,
$$

并且, 没有其他非平凡指数. 于是可得如下结论: 网络有 3 个不动点和 1 个长度为 9 的极限环. 找出 M 和 M^9 的对角非零列, 则可得到系统的三个不动点

$$
\begin{aligned}
X_1^* &= (1,0,1,0,0,0,1,0,0,0,1), \\
X_2^* &= (0,0,1,0,0,0,1,0,0,1,0), \\
X_3^* &= (0,0,1,0,0,0,1,0,0,0,0).
\end{aligned}
$$

系统的长度为 9 的极限环为

$$
\begin{aligned}
&(1,1,0,1,1,1,0,1,0,1,1) \to (0,1,0,1,1,1,0,1,1,1,1) \\
\to &(0,0,1,1,1,1,0,1,0,1,1) \to (0,0,0,1,1,1,1,1,1,1,1) \\
\to &(0,0,1,1,1,0,1,0,0,1,1) \to (1,0,0,1,0,0,1,0,0,1,1) \\
\to &(1,1,0,1,1,0,1,0,0,1,1) \to (1,1,0,1,1,0,0,0,0,1,1) \\
\to &(1,1,0,1,1,1,0,0,0,1,1) \to (1,1,0,1,1,1,0,1,0,1,1).
\end{aligned}
$$

M^k 的最小重复指数为 $M^{10} = M^{19}$, 过渡周期为 $T_t = 10$.

注 3.1.5 文献 [63] 指出系统有非平凡吸引子. 这里的结论指出, 系统有 3 个不动点和 1 个长度为 9 的极限环. 当 $X_{10} = X_{11} = X_4 = 1$ 时, 文献 [63] 与 [42] 均指出, 此时存在极限环, 这与本例结论相容. 当 $X_{10} = X_{11} = X_4 = 1$ 时, 不难验证系统的过渡周期仍为 $T_t = 10$. 但文献 [42] 称过渡周期 $T_t \leqslant 7$, 这是不对的. 考察如下轨线 $x(0) := x_0 = (0,1,1,1,1,0,0,0,0,1,1)$. 容易算出 $x(10) = (1,0,0,1,0,1,1,0,0,1,1)$, 它不属于极限环, 而 $x(11) = (1,1,0,1,1,0,1,0,\ 0,1,1)$ 属于极限环. 这说明 $T_t(x_0) \geqslant 10$.

下面这个例子出自文献 [54], 后在文献 [60] 中重新讨论过.

例 3.1.8　考察如下系统:

$$\begin{cases} X_1(t+1) = 1\bar{\vee}X_3(t)\bar{\vee}X_6(t)\bar{\vee}X_3(t)\wedge X_6(t), \\ X_2(t+1) = X_1(t), \\ X_3(t+1) = X_2(t), \\ X_4(t+1) = 1\bar{\vee}X_3(t)\bar{\vee}X_6(t)\bar{\vee}X_9(t)\bar{\vee}X_5(t)\wedge X_6(t)\bar{\vee}X_3(t)\wedge X_9(t) \\ \qquad\qquad\quad \bar{\vee}X_6(t)\wedge X_9(t)\bar{\vee}X_3(t)\wedge X_6(t)\wedge X_9(t), \\ X_5(t+1) = X_4(t), \\ X_6(t+1) = X_5(t), \\ X_7(t+1) = 1\bar{\vee}X_6(t)\bar{\vee}X_9(t)\bar{\vee}X_6(t)\wedge X_9(t), \\ X_8(t+1) = X_7(t), \\ X_9(t+1) = X_8(t). \end{cases} \tag{3.1.38}$$

记 $x_i = \vec{X}_i$, $i = 1, 2, \cdots, 9$, 上述系统的分量代数空间表达式为

$$\begin{cases} x_1(t+1) = M_n M_d x_3(t) x_6(t), \\ x_2(t+1) = x_1(t), \\ x_3(t+1) = x_2(t), \\ x_4(t+1) = M_c^2 M_n x_9(t) M_n x_3(t) M_n x_6(t), \\ x_5(t+1) = x_4(t), \\ x_6(t+1) = x_5(t), \\ x_7(t+1) = M_n M_d x_6(t) x_9(t), \\ x_8(t+1) = x_7(y), \\ x_9(t+1) = x_8(t). \end{cases} \tag{3.1.39}$$

记 $x(t) = \ltimes_{i=1}^{9} x_i(t)$, 系统的代数状态空间表达式为

$$x(t+1) = Mx(t),$$

这里

$$\begin{aligned} M =& M_n M_d (I_2 \otimes (I_2 \otimes (I_2 \otimes (I_2 \otimes M_c M_c M_n (I_2 \otimes M_n (I_2 \\ & \otimes M_n (I_2 \otimes (I_2 \otimes (I_2 \otimes M_n M_d)))))))))(I_2 \otimes W_{[2]}) W_{[2]} (I_4 \otimes W_{[2]}) \\ & \ltimes (I_2 \otimes W_{[2]})(I_{16} \otimes W_{[2]})(I_8 \otimes W_{[2]})(I_{64} \otimes W_{[2]})(I_{32} \otimes W_{[2]}) \end{aligned}$$

$$\ltimes (I_{16} \otimes W_{[2]})(I_{128} \otimes W_{[2]})(I_{64} \otimes W_{[2]})(I_{32} \otimes W_{[2]})(I_{128} \otimes W_{[2]})$$
$$\ltimes (I_{256} \otimes W_{[2]})(I_{1024} \otimes W_{[2]})(I_{512} \otimes W_{[2]})(I_{2048} \otimes W_{[2]})$$
$$\ltimes (I_{1024} \otimes W_{[2]})(I_2 \otimes (I_2 \otimes \mathrm{PR}_2(I_2 \otimes (I_2 \otimes (I_2$$
$$\otimes \mathrm{PR}_2\, \mathrm{PR}_2(I_2 \otimes (I_2 \otimes (I_2 \otimes \mathrm{PR}_2)))))))).$$

容易算出, 其非平凡指数为 $\mathrm{tr}(M^2) = 4$, 以及 $\mathrm{tr}(M^6) = 64$. 根据定理 3.1.2, 系统有两个长度为 2 的极限环和十个长度为 2 的极限环. 寻找 M^2 的对角非零列, 可得到两个长度为 2 的极限环

$$(1,0,1,1,0,1,1,0,1) \to (0,1,0,0,1,0,0,1,0) \to (1,0,1,1,0,1,1,0,1)$$
$$(1,0,1,0,0,0,0,1,0) \to (0,1,0,0,0,0,1,0,1) \to (1,0,1,0,0,0,0,1,0).$$

寻找 M^6 的对角非零列, 可得到十个长度为 6 的极限环

$$\begin{aligned}&(1,1,1,1,1,1,1,1,1) \to (0,1,1,0,1,1,0,1,1) \to (0,0,1,0,0,1,0,0,1)\\ &\to (0,0,0,0,0,0,0,0,0) \to (1,0,0,1,0,0,1,0,0) \to (1,1,0,1,1,0,1,1,0)\\ &\to (1,1,1,1,1,1,1,1,1).\end{aligned}$$

$$\begin{aligned}&(1,1,1,1,1,0,1,1,0) \to (0,1,1,0,1,1,1,1,1) \to (0,0,1,0,0,1,0,1,1)\\ &\to (0,0,0,0,0,0,0,0,1) \to (1,0,0,0,0,0,0,0,0) \to (1,1,0,1,0,0,1,0,0)\\ &\to (1,1,1,1,1,0,1,1,0).\end{aligned}$$

$$\begin{aligned}&(1,1,1,1,0,1,1,0,1) \to (0,1,1,0,1,0,0,1,0) \to (0,0,1,0,0,1,1,0,1)\\ &\to (0,0,0,0,0,0,0,1,0) \to (1,0,0,1,0,0,1,0,1) \to (1,1,0,0,1,0,0,1,0)\\ &\to (1,1,1,1,0,1,1,0,1).\end{aligned}$$

$$\begin{aligned}&(1,1,1,1,0,0,1,0,0) \to (0,1,1,0,1,0,1,1,0) \to (0,0,1,0,0,1,1,1,1)\\ &\to (0,0,0,0,0,0,0,1,1) \to (1,0,0,0,0,0,0,0,1) \to (1,1,0,0,0,0,0,0,0)\\ &\to (1,1,1,1,0,0,1,0,0).\end{aligned}$$

$$\begin{aligned}&(1,1,1,0,1,1,0,1,1) \to (0,1,1,0,0,1,0,0,1) \to (0,0,1,0,0,0,0,0,0)\\ &\to (0,0,0,0,0,0,1,0,0) \to (1,0,0,1,0,0,1,1,0) \to (1,1,0,1,1,0,1,1,1)\\ &\to (1,1,1,0,1,1,0,1,1).\end{aligned}$$

$$\begin{aligned}&(1,1,1,0,1,0,0,1,0) \to (0,1,1,0,0,1,1,0,1) \to (0,0,1,0,0,0,0,1,0)\\ &\to (0,0,0,0,0,0,1,0,1) \to (1,0,0,0,0,0,0,1,0) \to (1,1,0,1,0,0,1,0,1)\\ &\to (1,1,1,0,1,0,0,1,0).\end{aligned}$$

$$\begin{aligned}&(1,1,1,0,0,1,0,0,1) \to (0,1,1,0,0,0,0,0,0) \to (0,0,1,0,0,0,1,0,0)\\ &\to (0,0,0,0,0,0,1,1,0) \to (1,0,0,1,0,0,1,1,1) \to (1,1,0,0,1,0,0,1,1)\\ &\to (1,1,1,0,0,1,0,0,1).\end{aligned}$$

$$
\begin{aligned}
&(1,1,1,0,0,0,0,0,0) \to (0,1,1,0,0,0,1,0,0) \to (0,0,1,0,0,0,1,1,0)\\
&\to (0,0,0,0,0,0,1,1,1) \to (1,0,0,0,0,0,0,1,1) \to (1,1,0,0,0,0,0,0,1)\\
&\to (1,1,1,0,0,0,0,0,0).
\end{aligned}
$$

$$
\begin{aligned}
&(1,0,1,1,0,1,1,1,1) \to (0,1,0,0,1,0,0,1,1) \to (1,0,1,0,0,1,0,0,1)\\
&\to (0,1,0,0,0,0,0,0,0) \to (1,0,1,1,0,0,1,0,0) \to (0,1,0,0,1,0,1,1,0)\\
&\to (1,0,1,1,0,1,1,1,1).
\end{aligned}
$$

$$
\begin{aligned}
&(1,0,1,1,0,0,1,1,0) \to (0,1,0,0,1,0,1,1,1) \to (1,0,1,0,0,1,0,1,1)\\
&\to (0,1,0,0,0,0,0,0,1) \to (1,0,1,0,0,0,0,0,0) \to (0,1,0,0,0,0,1,0,0)\\
&\to (1,0,1,1,0,0,1,1,0).
\end{aligned}
$$

最后我们计算 M^k 的最小重复指数, 可知 $M^3 = M^9$. 因此, $T_t = 3$.

注 3.1.6　文献 [60] 指出, 上述网络没有不动点, 有两个长度为 2 的极限环, 这些结果与我们一致. [60] 还指出, 上述网络有六个长度为 6 的极限环, 这说明, 他们没有找到全部十个长度为 6 的极限环.

例 3.1.7 及例 3.1.8 说明了此前的逐一搜索的方法容易出错. 而定理 3.1.2 给出的一般性公式才是可靠的.

3.2　非齐次布尔网络

由式 (3.1.2) 定义的布尔网络称为齐次布尔网络, 它表示每个结点同时根据邻域结点上一时刻值更新自己的状态. 这类布尔网络也称马尔可夫型布尔网络, 这种布尔网络是最基本也最重要的一类布尔网络. 但也存在一些有用的非齐次布尔网络. 本节讨论这一类布尔网络.

3.2.1　不同步布尔网络

如果一个布尔网络, 它的结点在不同时刻更新自己的逻辑值, 则称其为不同步布尔网络. 下面考虑一种不同步布尔网络, 它的结点依顺序更新自己的逻辑值. 其网络方程为

$$
\begin{cases}
X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_{n-1}(t), X_n(t)),\\
X_2(t+1) = F_2(X_1(t+1), X_2(t), \cdots, X_{n-1}(t), X_n(t)),\\
\qquad\qquad\cdots\cdots\\
X_n(t+1) = F_n(X_1(t+1), \cdots, X_{n-1}(t+1), X_n(t)).
\end{cases}
\tag{3.2.1}
$$

这里, X_1 首先更新它的值. 于是, 当 X_2 更新它的值时, 它可以使用 X_1 更新后的值, 当 X_3 更新它的值时, 它可以使用 X_1 及 X_2 更新后的值, 其余类推.

考察系统 (3.2.1). 如果我们将第一个方程代入第二个方程, 则可得

$$\begin{aligned} X_2(t+1) &= F_2(F_1(X_1(t), X_2(t), \cdots, X_n(t)), X_2(t), \cdots, X_{n-1}(t), X_n(t)) \\ &:= \tilde{F}_2(X_1(t), X_2(t), \cdots, X_n(t)). \end{aligned} \tag{3.2.2}$$

这是一个标准的一步更新方程. 然后, 再将 F_1, $\tilde{F}_2$ 代入第三个方程, 可得 $\tilde{F}_3$, 等等. 最后, 我们得到

$$\begin{cases} X_1(t+1) = \tilde{F}_1(X_1(t), X_2(t), \cdots, X_{n-1}(t), X_n(t)), \\ X_2(t+1) = \tilde{F}_2(X_1(t), X_2(t), \cdots, X_{n-1}(t), X_n(t)), \\ \qquad \cdots\cdots \\ X_n(t+1) = \tilde{F}_n(X_1(t), \cdots, X_{n-1}(t), X_n(t)), \end{cases} \tag{3.2.3}$$

这里, $\tilde{F}_1 = F_1$.

(3.2.3) 称为 (3.2.1) 的齐次等价网络.

下面这个结论是显然的.

命题 3.2.1　(i) 顺序更新布尔网络 (3.2.1) 与其相同方程的齐次布尔网络有相同的不动点, 但极限环不同.

(ii) 顺序更新布尔网络 (3.2.1) 与其齐次等价网络 (3.2.3) 有相同的拓扑结构, 包括: 不动点、极限环, 以及每个吸引子的吸引域.

例 3.2.1　(i) 考察顺序更新布尔网络

$$\begin{cases} X_1(t+1) = X_2(t) \wedge X_3(t), \\ X_2(t+1) = \neg X_1(t+1), \\ X_3(t+1) = X_2(t+1) \vee X_3(t). \end{cases} \tag{3.2.4}$$

将第一个方程代入第二个方程, 可得

$$X_2(t+1) = \neg(X_2(t) \wedge X_3(t)) = \neg X_2(t) \vee \neg X_3(t).$$

将 $X_2(t+1)$ 代入第三个方程, 可得

$$X_3(t+1) = \neg X_2(t) \vee \neg X_3(t) \vee X_3(t) = 1.$$

合到一起则得

$$\begin{cases} X_1(t+1) = X_2(t) \wedge X_3(t), \\ X_2(t+1) = \neg X_2(t) \vee \neg X_3(t), \\ X_3(t+1) = 1. \end{cases} \tag{3.2.5}$$

记 $x_i(t) = \vec{X}_i(t)$, $i = 1, 2, 3$, 不难算得 (3.2.5) 的代数状态空间表达式为

$$x(t+1) = Mx(t), \tag{3.2.6}$$

这里

$$M = \delta_8[3, 5, 5, 5, 3, 5, 5, 5].$$

其状态转移图见图 3.2.1.

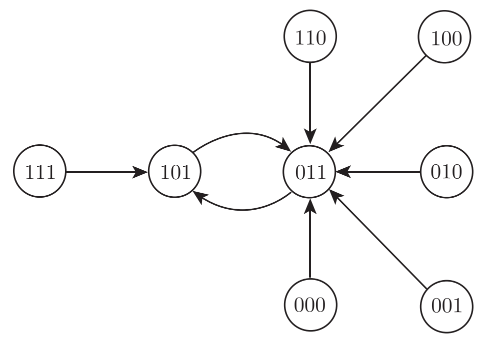


图 3.2.1　网络 (3.2.4) 的状态转移图

(ii) 考察与 (3.2.4) 相同方程的齐次布尔网络

$$\begin{cases} X_1(t+1) = X_2(t) \wedge X_3(t), \\ X_2(t+1) = \neg X_1(t), \\ X_3(t+1) = X_2(t) \vee X_3(t). \end{cases} \tag{3.2.7}$$

则其代数状态空间表达式为

$$x(t+1) = Nx(t), \tag{3.2.8}$$

这里

$$N = \delta_8[3, 7, 7, 8, 1, 5, 5, 6].$$

其状态转移图见图 3.2.2.

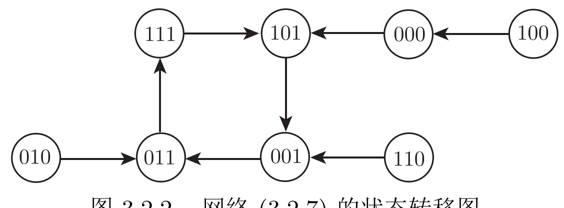


图 3.2.2　网络 (3.2.7) 的状态转移图

比较图 3.2.1 与图 3.2.2 可知, 网络 (3.2.4) 与网络 (3.2.7) 极限环不同.

3.2.2 高阶布尔网络

本节讨论高阶布尔网络, 它是另一种非齐次布尔网络. 本节的内容部分来自文献 [85].

定义 3.2.1 一个布尔网络称为 μ 阶布尔网络, 如果结点的当前状态决定于过去 μ 个时刻的状态. 严格地说, 其动态演化方程为

$$\begin{cases} X_1(t+1) = F_1(X_1(t-\mu+1),\cdots,X_n(t-\mu+1),\cdots,X_1(t),\cdots,X_n(t)), \\ X_2(t+1) = F_2(X_1(t-\mu+1),\cdots,X_n(t-\mu+1),\cdots,X_1(t),\cdots,X_n(t)), \\ \qquad\qquad\cdots\cdots \\ X_n(t+1) = F_n(X_1(t-\mu+1),\cdots,X_n(t-\mu+1),\cdots,X_1(t),\cdots,X_n(t)), \\ t \geqslant \mu-1, \end{cases} \tag{3.2.9}$$

这里, $F_i:\mathcal{D}^{\mu n}\to\mathcal{D}$, $i=1,\cdots,n$ 为逻辑函数.

与普通高阶差分方程一样, 要确定 μ 阶布尔网络的解轨线, 需要 μ 个时刻的初始值, 即

$$X_i(j) = a_{ij}, \quad i=1,\cdots,n; \quad j=0,\cdots,\mu-1. \tag{3.2.10}$$

我们用一个例子来描述高阶布尔网络. 这个例子来自生物网络, 用于描述细胞网络中的耦合振子[54].

例 3.2.2 考察下列布尔网络:

$$\begin{cases} X_1(t+3) = \neg(X_1(t)\wedge X_2(t+1)), \\ X_2(t+3) = \neg(X_1(t+1)\wedge X_2(t)). \end{cases} \tag{3.2.11}$$

不难将其转化为高阶布尔网络 (3.2.9) 的形式.

$$\begin{cases} X_1(t+1) = \neg(X_1(t-2)\wedge X_2(t-1)), \\ X_2(t+1) = \neg(X_1(t-1)\wedge X_2(t-2)), \end{cases} \quad t \geqslant 2. \tag{3.2.12}$$

这是 3 阶布尔网络.

下面这个例子来自文献 [103]. 它虽然不是逻辑系统, 而是一个博弈问题, 但如果每个玩家的策略为 $X_i\in\mathcal{D}$, 则其数学模型也可表示为布尔网络系统.

例 3.2.3 考察重复囚徒困境. 设玩家 1 是机器, 玩家 2 是人. 策略集为

$0:$ 玩家与对手合作;
$1:$ 玩家不与对手合作.

用 $\{X_1(0), X_1(1), \cdots\}$ 表示机器的策略; $\{X_2(0), X_2(1), \cdots\}$ 表示人的策略. 设机器有长度为 μ 的记忆, 即, 它的策略由双方此前的 μ 个记录确定. 那么, 机器的策略可表示为

$$X_1(t+1) = F_1(X_1(t-\mu+1), X_2(t-\mu+1), \cdots, X_1(t), X_2(t)). \tag{3.2.13}$$

文献 [103] 指出, 人的最佳应对策略也是使用长度为 μ 的记忆, 即

$$X_2(t+1) = F_2(X_1(t-\mu+1), X_2(t-\mu+1), \cdots, X_1(t), X_2(t)). \tag{3.2.14}$$

将 (3.2.13) 与 (3.2.14) 放到一起, 即得

$$\begin{cases} X_1(t+1) = f_1(X_1(t-\mu+1), X_2(t-\mu+1), \cdots, X_1(t), X_2(t)), \\ X_2(t+1) = f_2(X_1(t-\mu+1), X_2(t-\mu+1), \cdots, X_1(t), X_2(t)). \end{cases} \tag{3.2.15}$$

这是一个标准的 μ 阶布尔网络.

类似于齐次布尔网络, 我们也希望知道高阶布尔网络的拓扑结构. 由于 μ 阶布尔网络的轨线是由长度为 μ 的初始值决定的, 我们必须对它的不动点和极限环重新定义.

定义 3.2.2　考察网络 (3.2.9). 记其状态空间为

$$\mathcal{X} = \{X \mid X = (x_1, \cdots, x_n) \in D^n\}.$$

$X(t) \in \mathcal{X}$ 为系统在 t 时刻的状态, 即 $X(t) = (X_1(t), X_2(t), \cdots, X_n(t))$.

(i)

$$(X(t), X(t+1), \cdots, X(t+\mu))$$

称为一个不动点, 如果

$$(X(t), X(t+1), \cdots, X(t+\mu)) = (X(t+1), X(t+2), \cdots, X(t+\mu+1)). \tag{3.2.16}$$

(ii)

$$(X(t), X(t+1), \cdots, X(t+\mu), \cdots, X(t+s), \cdots, X(t+s+\mu))$$

称为一个长度为 s 的极限环, 如果

$$\begin{aligned} &(X(t+p), X(t+p+1), \cdots, X(t+p+\mu)) \\ &\neq (X(t+q), X(t+1+q), \cdots, X(t+\mu+q)), \quad t \geqslant p < q < s, \end{aligned} \tag{3.2.17}$$

并且,

$$(X(t), X(t+1), \cdots, X(t+\mu)) = (X(t+s), X(t+s+1), \cdots, X(t+s+\mu)). \tag{3.2.18}$$

类似于齐次布尔网络, 我们希望能将高阶布尔网络从逻辑表达式转化为代数表达式, 从而计算出其不动点、极限环等拓扑要素.

3.2.3 高阶布尔网络的第一代数表达式

记 $x_i(t) = \vec{X}_i(t)$, $i = 1, 2, \cdots, n$. 定义

$$\begin{cases} x(t) = \ltimes_{i=1}^{n} x_i(t) \in \Delta_{2^n} \\ z(t) = \ltimes_{i=t}^{t+\mu-1} x(i) \in \Delta_{2^{\mu n}}, \quad t = 0, 1, \cdots. \end{cases}$$

设 F_i 的结构矩阵为 $M_i \in \mathcal{L}_{2\times 2^{\mu n}}$. 则 (3.2.9) 有其分量代数状态空间表达式

$$x_i(t+1) = M_i z(t-\mu+1), \quad i = 1, \cdots, n; \quad t = \mu-1, \mu, \mu+1, \cdots. \tag{3.2.19}$$

利用 (3.2.19) 可得 (3.2.9) 代数状态空间表达式

$$x(t+1) = Mz(t-\mu+1), \quad t \geqslant \mu, \tag{3.2.20}$$

这里

$$M = M_1 * M_2 * \cdots * M_n. \tag{3.2.21}$$

根据定义并利用 (3.2.20), 不难得到如下等式:

$$\begin{aligned} z(t+1) &= \ltimes_{i=t+1}^{t+\mu} x(i) \\ &= \mathbf{1}_{2^n}^{\mathrm{T}} \ltimes_{i=t}^{t+\mu} x(i) \\ &= \mathbf{1}_{2^n}^{\mathrm{T}} \ltimes_{i=t}^{t+\mu-1} x(i) x(t+\mu) \\ &= \mathbf{1}_{2^n}^{\mathrm{T}} z(t) M z(t) \\ &:= Lz(t), \end{aligned} \tag{3.2.22}$$

这里

$$L = \mathbf{1}_{2^n}^{\mathrm{T}} (I_{2^{\mu n}} \otimes M)\, \mathrm{PR}_{\mu n}. \tag{3.2.23}$$

式 (3.2.22) 称为高阶布尔网络的第一代数形式.

下面给一个例子.

例 3.2.4

$$\begin{cases} X_1(t+1) = X_3(t-1) \vee (X_1(t) \wedge X_2(t)), \\ X_2(t+1) = \neg(X_3(t-1) \wedge X_1(t)), \\ X_3(t+1) = X_2(t-1) \wedge X_2(t). \end{cases} \tag{3.2.24}$$

记 $x_i(t) = \vec{X}_i$, $i = 1,2,3$. 可得

$$\begin{cases} x_1(t+1) = M_d x_3(t-1) M_c x_1(t) x_2(t), \\ x_2(t+1) = M_n M_c x_3(t-1) x_1(t), \\ x_3(t+1) = M_c x_2(t-1) x_2(t). \end{cases} \tag{3.2.25}$$

记 $x(t) = x_1(t)x_2(t)x_3(t)$. 则 (3.2.25) 可变为 (3.2.24) 的分量代数状态空间方法表达式:

$$\begin{cases} x_1(t+1) = M_1 x(t-1) x(t), \\ x_2(t+1) = M_2 x(t-1) x(t), \\ x_3(t+1) = M_3 x(t-1) x(t), \end{cases} \tag{3.2.26}$$

这里

$$\begin{aligned} M_1 = \delta_4[&1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,\\ &1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,\\ &1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,\\ &1,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2];\\ M_2 = \delta_4[&2,2,2,2,1,1,1,1,1,1,1,1,1,1,1,1,\\ &2,2,2,2,1,1,1,1,1,1,1,1,1,1,1,1,\\ &2,2,2,2,1,1,1,1,1,1,1,1,1,1,1,1,\\ &2,2,2,2,1,1,1,1,1,1,1,1,1,1,1,1];\\ M_3 = \delta_4[&1,1,2,2,1,1,2,2,1,1,2,2,1,1,2,2,\\ &2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,\\ &1,1,2,2,1,1,2,2,1,1,2,2,1,1,2,2,\\ &2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2]. \end{aligned}$$

利用分量式可得 (3.2.24) 的代数状态空间表达式:

$$x(t+1) = Mx(t-1)x(t), \tag{3.2.27}$$

这里

$$M = \delta_8[3,3,4,4,1,1,2,2,1,1,6,6,5,5,6,6,\\ 4,4,4,4,2,2,2,2,2,2,6,6,6,6,6,6,\\ 3,3,4,4,1,1,2,2,1,1,6,6,5,5,6,6,\\ 4,4,4,4,2,2,2,2,2,2,6,6,6,6,6,6].$$

令 $z(t) = x(t)x(t+1)$, $t \geqslant 1$, 最后可得

$$\begin{aligned} z(t+1) &= x(t+1)x(t+2) = \mathbf{1}_8^{\mathrm{T}} x(t)x(t+1)x(t+2) \\ &= \mathbf{1}_8^{\mathrm{T}} x(t)x(t+1)Mx(t)x(t+1) \\ &= \mathbf{1}_8^{\mathrm{T}} (I_{2^6} \otimes M)\,\mathrm{PR}_{2^6}\, x(t)x(t+1) \\ &:= Lz(t), \end{aligned} \tag{3.2.28}$$

这里

$$L = \delta_{64}[3,11,20,28,33,41,50,58,1,9,22,30,37,45,54,62,\\ 4,12,20,28,34,42,50,58,2,10,22,30,38,46,54,62,\\ 3,11,20,28,33,41,50,58,1,9,22,30,37,45,54,62,\\ 4,12,20,28,34,42,50,58,2,10,22,30,38,46,54,62]. \tag{3.2.29}$$

我们需要证明: 高阶布尔网络 (3.2.20) 与齐次布尔网络 (3.2.22) 具有相同的拓扑结构, 包括不动点、极限环、过渡周期及吸引域等. 这个等价性可以使高阶布尔网络的相关拓扑结构可计算.

引理 3.2.1 网络 (3.2.20) 的轨线与网络 (3.2.22) 的轨线一一对应.

证明 分别记网络 (3.2.20) 和网络 (3.2.22) 为 T_x 和 T_z. 注意到网络轨线是由初值唯一决定的, (3.2.20) 的初值是 $\{x(0), x(1), \cdots, x(\mu-1)\}$, (3.2.22) 的初值是 $z(0)$. 令 $z(0) = \ltimes_{i=0}^{\mu-1} x(i)$, 那么, 不难推出

$$z(t) = \ltimes_{i=t}^{t+\mu-1} x(i), \quad t \geqslant 0, \tag{3.2.30}$$

即 T_z 由 T_x 唯一确定. 反之,

$$x(t) = (I_{2^n} \otimes \mathbf{1}_{2^{(\mu-1)n}}^{\mathrm{T}}) z(t), \quad t \geqslant \mu, \tag{3.2.31}$$

即 T_x 由 T_z 唯一确定.

故 T_z 与 T_x 一一对应. □

实际上, (3.2.30) 定义了一个映射 $\phi : T_x \to T_z$, 同时, (3.2.31) 定义了一个映射 $\pi : T_z \to T_x$. 利用后者, 我们即可计算高阶布尔网络的吸引子.

定理 3.2.1 (i) 设 z_* 为 (3.2.22) 的一个不动点, 则 $x_* = \pi(z_*)$ 为 (3.2.20) 的一个不动点. 反之, 设 x_* 为 (3.2.20) 的一个不动点, 则 $z_* = \phi(z_*)$ 为 (3.2.22) 的一个不动点. 这里,

$$z_* = \ltimes_{i=t}^{t+\mu-1} x(i),$$

并且, $x(t) = x_*$.

(ii) 设 $(z_0, z_1, \cdots, z_p = z_0)$ 为 (3.2.22) 的一个长度为 p 的极限环, 则 $(\pi(z_0), \pi(z_1), \cdots, \pi(z_p) = \pi(z_0))$ 为 (3.2.20) 的一个长度为 p 的极限环.

设 $(x_0, x_1, \cdots, x_p = x_0)$ 为 (3.2.20) 的一个长度为 p 的极限环, 则 $(\phi(x_0), \phi(x_1), \cdots, \phi(x_p) = \phi(x_0))$ 为 (3.2.22) 的一个长度为 p 的极限环.

(iii) 设 $x = \pi(z)$, 则 x 的过渡周期与 z 的过渡周期相等. 因此, 网络 (3.2.20) 的过渡周期与网络 (3.2.22) 的过渡周期也相等.

定理 3.2.1 指出, 要求出网络 (3.2.20) 的不动点与极限环等, 只要求出其等价的齐次网络 (3.2.22) 的不动点与极限环等就够了. 因此, 此前发展的齐次网络不动点和极限环等的计算方法就可以用来计算高阶网络了.

下面考虑几个例子.

例 3.2.5 回忆例 3.2.2. 记 $x(t) = \vec{X}_1(t)\vec{X}_2(t)$. 则不难得到 (3.2.11) 的代数状态空间表示

$$x(t+1) = Mx(t-2)x(t-1)x(t), \tag{3.2.32}$$

这里

$$\begin{aligned} M = \delta_4[&4,4,4,4,2,2,2,2,3,3,3,3,1,1,1,1,\\ &3,3,3,3,1,1,1,1,3,3,3,3,1,1,1,1,\\ &2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1,\\ &1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]. \end{aligned}$$

令 $z(t) = x(t)x(t+1)x(t+2)$. 则得

$$\begin{aligned} z(t+1) &= x(t+1)x(t+2)x(t+3)\\ &= \mathbf{1}_4^{\mathrm{T}} x(t)x(t+1)x(t+2)x(t+3)\\ &= \mathbf{1}_4^{\mathrm{T}} x(t)x(t+1)x(t+2)Mx(t)x(t+1)x(t+2)\\ &= \mathbf{1}_4^{\mathrm{T}} (I_{2^6}\otimes M)\,\mathrm{PR}_{2^6}\, x(t)x(t+1)x(t+2)\\ &:= Lz(t), \end{aligned} \tag{3.2.33}$$

这里

$$L = \delta_{2^6}[4, 8, 12, 16, 18, 22, 26, 30, 35, 39, 43, 47, 49, 53, 57, 61,$$
$$3, 7, 11, 15, 17, 21, 25, 29, 35, 39, 43, 47, 49, 53, 57, 61,$$
$$2, 6, 10, 14, 18, 22, 26, 30, 33, 37, 41, 45, 49, 53, 57, 61,$$
$$1, 5, 9, 13, 17, 21, 25, 29, 33, 37, 41, 45, 49, 53, 57, 61].$$

要找出 (3.2.10) 的不动点和极限环, 只要找 (3.2.33) 的相关量就行了. 直接计算 $\mathrm{tr}(L^k), k = 1, 2, \cdots, 64$, 并检查非平凡指数, 可以得到

$$\mathrm{tr}(L^2) = 2, \quad \mathrm{tr}(L^5) = 5, \quad \mathrm{tr}(L^{10}) = 17.$$

根据定理 3.1.1 和定理 3.1.2, 我们知道, 网络 (3.2.10) 没有不动点; 它有一个长度为 2 的极限环、一个长度为 5 的极限环和一个长度为 10 的极限环. 下面寻找这几个极限环.

(i) 长度为 2 的极限环.

先找 (3.2.33) 长度为 2 的极限环. 考察 L^2. 容易找出, 它的第 26 列对角元不为零, 即

$$\mathrm{Col}_{26}(L^2) = \delta_{64}^{26}.$$

利用它可计算极限环如下: $L\delta_{64}^{26} = \delta_{64}^{29}$, $L\delta_{64}^{29} = \delta_{64}^{26}$, 于是可得 (3.2.33) 长度为 2 的极限环如下:

$$\delta_{64}^{26} \to \delta_{64}^{29} \to \delta_{64}^{26}.$$

于是 (3.2.10) 长度为 2 的极限环如下:

$$\pi\left(\delta_{64}^{26}\right) \to \pi\left(\delta_{64}^{29}\right) \to \pi\left(\delta_{64}^{26}\right) = \delta_4^2 \to \delta_4^3 \to \delta_4^2.$$

返回到逻辑值, 即为 $(1, 0) \to (0, 1) \to (1, 0)$, 图 3.2.3 画出这一极限环.

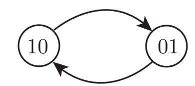


图 3.2.3 (3.2.10) 长度为 2 的极限环

(ii) 长度为 5 的极限环.

先找 (3.2.33) 长度为 5 的极限环. 考察 L^5. 它的第 1 列对角元不为零, 即

$$\mathrm{Col}_1(L^2) = \delta_{64}^1.$$

依次可算出 δ_{64}^1, $L\delta_{64}^1 = \delta_{64}^4$, $L^2\delta_{64}^1 = \delta_{64}^{16}$, $L^3\delta_{64}^1 = \delta_{64}^{61}$, $L^4\delta_{64}^1 = \delta_{64}^{49}$. 因此, 网络 (3.2.10) 长度为 5 的极限环如下:

$$\pi(\delta_{64}^1) \to \pi(\delta_{64}^4) \to \pi(\delta_{64}^{16}) \to \pi(\delta_{64}^{61}) \to \pi(\delta_{64}^{49}) \to \pi(\delta_{64}^1),$$

即

$$\delta_4^1 \to \delta_4^1 \to \delta_4^1 \to \delta_4^4 \to \delta_4^4 \to \delta_4^1.$$

返回逻辑值, 网络 (3.2.10) 长度为 5 的极限环见图 3.2.4.

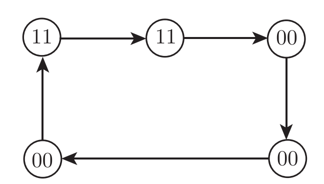


图 3.2.4　(3.2.10) 长度为 5 的极限环

(iii) 长度为 10 的极限环.

先找 (3.2.33) 长度为 10 的极限环. 考察 L^{10}. 它的第 2 列对角元不为零, 即

$$\mathrm{Col}_2(L^{10}) = \delta_{64}^2.$$

于是, 极限环为 δ_{64}^2, $L\delta_{64}^2 = \delta_{64}^8$, $L^2\delta_{64}^2 = \delta_{64}^{30}$, $L^3\delta_{64}^2 = \delta_{64}^{53}$, $L^4\delta_{64}^2 = \delta_{64}^{17}$, $L^5\delta_{64}^2 = \delta_{64}^3$, $L^6\delta_{64}^2 = \delta_{64}^{12}$, $L^7\delta_{64}^2 = \delta_{64}^{47}$, $L^8\delta_{64}^2 = \delta_{64}^5$, $L^9\delta_{64}^2 = \delta_{64}^{33}$. 于是 (3.2.10) 的长度为 10 的极限环为

$$\begin{aligned}&\pi(\delta_{64}^2) \to \pi(\delta_{64}^8) \to \pi(\delta_{64}^{30}) \to \pi(\delta_{64}^{53}) \to \pi(\delta_{64}^{17}) \to \pi(\delta_{64}^3)\\ \to &\pi(\delta_{64}^{12}) \to \pi(\delta_{64}^{47}) \to \pi(\delta_{64}^5) \to \pi(\delta_{64}^{33}) \to \pi(\delta_{64}^2),\end{aligned}$$

即

$$\delta_4^1 \to \delta_4^1 \to \delta_4^2 \to \delta_4^4 \to \delta_4^2 \to \delta_4^1 \to \delta_4^1 \to \delta_4^3 \to \delta_4^4 \to \delta_4^3 \to \delta_4^1.$$

图 3.2.5 画出这一极限环.

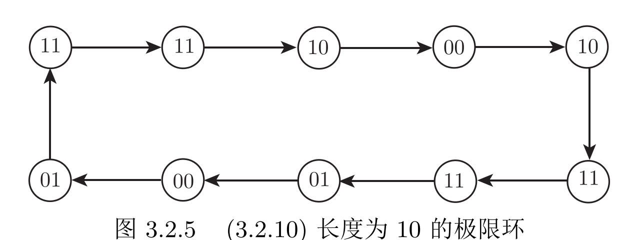


图 3.2.5　(3.2.10) 长度为 10 的极限环

以上的结果与文献 [60] 相符.

例 3.2.6　回忆例 3.2.4. 寻找 (3.2.24) 的吸引子. 考察其等价多次网络 (3.2.28). 计算 $\mathrm{tr}(L^k), k = 1, 2, \cdots, 64$, 易知

$$\mathrm{tr}(L^k) = \begin{cases} 8, & k = 8i,\ i = 1, 2, \cdots, \\ 0, & \text{其他}. \end{cases}$$

根据定理 3.1.1 和定理 3.1.2, 我们知道 (3.2.28) 只有一个长度为 8 的极限环. 选择对角元非零列 $\mathrm{Col}_3(L^8)=\delta_{64}^3$, 则可生成长度为 8 的极限环:

$$\begin{aligned} &z=\delta_{64}^3, \quad && Lz=\delta_{64}^{20}, \quad && L^2z=\delta_{64}^{28}, \quad && L^3z=\delta_{64}^{30},\\ &L^4z=\delta_{64}^{46}, && L^5z=\delta_{64}^{45}, && L^6z=\delta_{64}^{37}, && L^7z=\delta_{64}^{33}. \end{aligned}$$

注意到 $L^8z=z=\delta_{64}^3$, 我们得到 (3.2.28) 的长度为 8 的极限环:

$$\delta_{64}^3\to\delta_{64}^{20}\to\delta_{64}^{28}\to\delta_{64}^{30}\to\delta_{64}^{46}\to\delta_{64}^{45}\to\delta_{64}^{37}\to\delta_{64}^{33}\to\delta_{64}^3.$$

利用投影

$$\pi(z)=\left(I_8\otimes\mathbf{1}_8^{\mathrm{T}}\right)z,$$

可得 (3.2.24) 的极限环

$$\begin{aligned} &\pi(\delta_{64}^3)\to\pi(\delta_{64}^{20})\to\pi(\delta_{64}^{28})\to\pi(\delta_{64}^{30})\to\pi(\delta_{64}^{46})\\ \to&\pi(\delta_{64}^{45})\to\pi(\delta_{64}^{37})\to\pi(\delta_{64}^{33})\to\pi(\delta_{64}^3), \end{aligned}$$

即

$$\delta_8^1\to\delta_8^3\to\delta_8^4\to\delta_8^4\to\delta_8^6\to\delta_8^6\to\delta_8^5\to\delta_8^5\to\delta_8^1.$$

返回逻辑值, 则 (3.2.24) 的极限环可用图 3.2.6 描述.

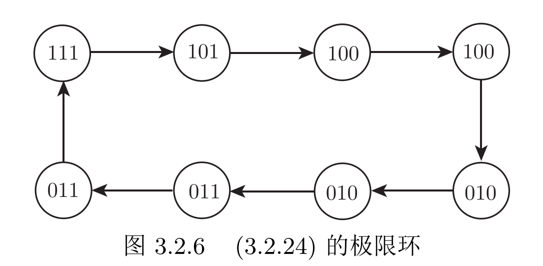


图 3.2.6 (3.2.24) 的极限环

不难计算, (3.2.28) 的过渡周期为 7. 由定理 3.2.1 可知, (3.2.24) 的过渡周期亦为 7. 因此, 从任何一个状态 $(X_1(t_0),X_2(t_0))$ 出发, 也不管 $(X_1(t_0-1),X_2(t_0-1))$ 的值, 至多 7 步, 状态必定进入极限环.

注 3.2.1 在以上例子中, 我们没有考虑吸引子 (包括不动点与极限环) 的吸引域. 实际上, 对于高阶系统吸引域不是一个定义好的概念. 因为, 从一个点 $x(0)$ 出发, 轨线可能进入不同的吸引子, 因此, 轨线不由 $x(0)$ 完全确定.

例如, 我们回顾例 3.2.5 (亦即例 3.2.2). 网络 (3.2.11) 有 3 个极限环, 长度分别为 1, 2, 5. 利用代数表达式 (3.2.33), 分别从初态 δ_{64}^{60} 以及 δ_{64}^{50} 出发, 有

$$\delta_{64}^{60}\to\delta_{64}^{45}\to\left\{\delta_{64}^{49}\to\delta_{64}^{1}\to\delta_{64}^{4}\to\delta_{64}^{16}\to\delta_{64}^{61}\to\delta_{64}^{49}\right\},$$

$$\delta_{64}^{50} \to \delta_{64}^{5} \to \delta_{64}^{18} \to \delta_{64}^{7} \to \left\{\delta_{64}^{26} \to \delta_{64}^{39} \to \delta_{64}^{26}\right\}.$$

回到网络 (3.2.33), 根据定理 3.2.1 可知, 从初态 $\pi(\delta_{64}^{60})$ 以及 $\pi(\delta_{64}^{50})$ 出发, 我们有轨线

$$\pi(\delta_{64}^{60}) \to \pi(\delta_{64}^{45}) \to \left\{\pi(\delta_{64}^{49}) \to \pi(\delta_{64}^{1}) \to \pi(\delta_{64}^{4}) \to \pi(\delta_{64}^{16}) \to \pi(\delta_{64}^{61}) \to \pi(\delta_{64}^{49})\right\},$$

$$\pi(\delta_{64}^{50}) \to \pi(\delta_{64}^{5}) \to \pi(\delta_{64}^{18}) \to \pi(\delta_{64}^{7}) \to \left\{\pi(\delta_{64}^{26}) \to \pi(\delta_{64}^{39}) \to \pi(\delta_{64}^{26})\right\}.$$

注意到

$$\pi(\delta_{64}^{60}) = \pi(\delta_{64}^{50}) = \delta_4^4,$$

也就是说, 对于网络 (3.2.33), 从初态 δ_4^4 出发, 我们有

$$\delta_4^4 \to \delta_4^3 \to \left\{\delta_4^4 \to \delta_4^1 \to \delta_4^1 \to \delta_4^1 \to \delta_4^4 \to \delta_4^4\right\},$$

$$\delta_4^4 \to \delta_4^1 \to \delta_4^2 \to \delta_4^1 \to \left\{\delta_4^2 \to \delta_4^3 \to \delta_4^2\right\}.$$

由此可见, δ_4^4 可以进入两个不同的极限环.

3.2.4 高阶布尔网络的第二代数表达式

定义

$$w(\tau) := x(\mu\tau)x(\mu\tau+1)\cdots x(\mu\tau+(\mu-1)) = z(\mu\tau). \tag{3.2.34}$$

那么, 我们有

$$w(\tau+1) = z(\mu\tau+\mu) = L^{\mu}z(\mu\tau) = L^{\mu}w(\tau),$$

这里 (见 (3.2.23))

$$L = \mathbf{1}_{2^n}^{\mathrm{T}}(I_{2^{\mu n}} \otimes M)\,\mathrm{PR}_{\mu n}.$$

因此可得

$$w(\tau+1) = \Gamma w(\tau), \tag{3.2.35}$$

这里

$$\Gamma = \left[\mathbf{1}_{2^n}^{\mathrm{T}}(I_{2^{\mu n}} \otimes M)\,\mathrm{PR}_{\mu n}\right]^{\mu},$$

其初值为

$$w(0) = \ltimes_{i=0}^{\mu-1} x(i). \tag{3.2.36}$$

(3.2.35)—(3.2.36) 称为 μ 阶布尔网络 (3.2.9) 的第二代数表达式.

实际上, 通过重新设定采样周期, 第二代数表达式的变量 $w(\tau)$, $\tau = 0, 1, \cdots$, 是无重复的每组 μ 个连续 $x(t)$ 的乘积. 因此, 显然网络 (3.2.9) 与 (3.2.35) 的轨线有一一对应关系.

命题 3.2.2 (3.2.9) 的轨线与 (3.2.35) 的轨线一一对应, 对应关系由下式决定:

$$w(\tau)=x(\mu\tau)x(\mu\tau+1)\cdots x(\mu\tau+(\mu-1)),\quad \tau=0,1,\cdots. \tag{3.2.37}$$

由命题 3.2.2 可见, 用第二代数表达式计算高阶布尔网络的轨线是很方便的. 但是, 它却很难用于计算高阶布尔网络的不动点及极限环等. 因此, 它与第一代数表达式各有所长. 我们可根据目的来选择它们.

命题 3.2.3 设网络 (3.2.9) 有一个长度为 α 的极限环. 记 α 与 μ 的最小公倍数为 $\beta=\mathrm{lcm}(\alpha,\mu)$. 那么, 网络 (3.2.35) 有一个长度为 $\gamma=\beta/\mu$ 的极限环.

证明 设 $s>0$ 足够大, 使 $x((s-1)\mu+1)$ 在长度为 α 的极限环上. 因为 β 是 α 的倍数, 我们有

$$\begin{cases}x((s-1)\mu+1)=x((s-1)\mu+1+\beta),\\ x((s-1)\mu+2)=x((s-1)\mu+2+\beta),\\ \qquad\qquad\cdots\cdots\\ x(s\mu)=x(s\mu+\beta).\end{cases} \tag{3.2.38}$$

将 (3.2.38) 两边各式相乘 (做半张量积乘法) 得

$$w(s)=w\left(s+\frac{\beta}{\mu}\right)=w(s+\gamma). \tag{3.2.39}$$

不难检验 γ 是让 (3.2.39) 成立的最小正整数. □

例 3.2.7 回忆例 3.2.4. 由式 (3.2.28), 可知

$$z(t+1)=Lz(t),$$

这里, L 由 (3.2.29) 给出.

令 $w(\tau)=x(2\tau)x(2\tau+1)=z(2\tau)$, 那么 (3.2.24) 的第二代数表达式为

$$w(\tau+1)=z(2\tau+2)=L^2z(2\tau)=L^2w(\tau)=\Gamma w(\tau), \tag{3.2.40}$$

这里

$$\begin{aligned}\Gamma=\delta_{64}[&20,22,28,30,3,1,12,10,3,1,42,46,33,37,42,46,\\ &28,30,28,30,11,9,12,10,11,9,42,46,41,45,42,46,\\ &20,22,28,30,3,1,12,10,3,1,42,46,33,37,42,46,\\ &28,30,28,30,11,9,12,10,11,9,42,46,41,45,42,46].\end{aligned}$$

为发现 (3.2.40) 的吸引子, 计算 $\mathrm{tr}(\Gamma^s), s=1,2,\cdots$ 并找出非平凡指数 s. 不难验证

$$\mathrm{tr}(\Gamma^s)=\begin{cases}8, & s=4i, i=1,2,\cdots,\\ 0, & 其他.\end{cases}$$

根据定理 3.1.1 和定理 3.1.2 可知, (3.2.40) 有两个长度为 4 的极限环.

考虑 Γ^4.

(i) $\mathrm{Col}_3(\Gamma^4)=\delta_{64}^3$. 利用 δ_{64}^3 可得 $\Gamma\delta_{64}^3=\delta_{64}^{28}$, $\Gamma\delta_{64}^{28}=\delta_{64}^{46}$, $\Gamma\delta_{64}^{46}=\delta_{64}^{37}$, 以及 $\Gamma\delta_{64}^{37}=\delta_{64}^3$, 这是一个长度为 4 的极限环. 返回到逻辑值, 可得

$$(1,1,1,1,0,1)\to(1,0,0,1,0,0)\to(0,1,0,0,1,0)\to(0,1,1,0,1,1)\to\cdots. \tag{3.2.41}$$

(ii) $\mathrm{Col}_{20}(\Gamma^4)=\delta_{64}^{20}$. 利用 δ_{64}^{20} 可得 $\Gamma\delta_{64}^{20}=\delta_{64}^{30}$, $\Gamma\delta_{64}^{30}=\delta_{64}^{45}$, $\Gamma\delta_{64}^{45}=\delta_{64}^{33}$, 以及 $\Gamma\delta_{64}^{33}=\delta_{64}^{20}$. 这给出了另一个长度为 4 的极限环:

$$(1,0,1,1,0,0)\to(1,0,0,0,1,0)\to(0,1,0,0,1,1)\to(0,1,1,1,1,1)\to\cdots. \tag{3.2.42}$$

注意到 (3.2.24) 的 $\mu=2$, 有长度为 $\beta=8$ 的极限环. 命题 3.2.3 断言其第二代数表达式有长度 $\gamma=\beta/\mu=4$ 的极限环. 以上计算验证了这一断言.

3.3 k 值逻辑与混合值逻辑

定义 3.3.1 考虑一个逻辑动态系统

$$\begin{cases}X_1(t+1)=F_1(X_1(t),X_2(t),\cdots,X_n(t)),\\ X_2(t+1)=F_2(X_1(t),X_2(t),\cdots,X_n(t)),\\ \qquad\qquad\cdots\cdots\\ X_n(t+1)=F_n(X_1(t),X_2(t),\cdots,X_n(t)).\end{cases} \tag{3.3.1}$$

(i) 如果 $X_i(t)\in\mathcal{D}_k$, $F_i:\mathcal{D}_k^n\to\mathcal{D}_k$ 为 k 值逻辑函数, $i=1,2,\cdots,n$, 则 (3.3.1) 称为 k 值逻辑网络.

(ii) 如果 $X_i(t)\in\mathcal{D}_{k_i}$, $F_i:\prod_{j=1}^k\mathcal{D}_{k_j}\to\mathcal{D}_{k_i}$ 为混合值逻辑函数, $i=1,2,\cdots,n$, 则 (3.3.1) 称为混合值逻辑网络.

注 3.3.1 (i) 这里所说的 k 值 (或混合值) 逻辑是在广义意义下的“逻辑”系统, 即变量 $X_i(t)$ 取 k 个不同的值就可称为 k 值逻辑系统. 不必要求 $X_i(t)$ 确实

具有逻辑的含义. 例如考虑例 3.2.3, 这里, 变量 (策略) 可取“合作”“不合作”这两种值, 这里策略应不存在逻辑意义, 但它只取两个值, 故可看作布尔网络.

(ii) 在严格逻辑意义下

$$\mathcal{D}_k = \left\{0, \frac{1}{k-1}, \cdots, \frac{k-2}{k-1}, 1\right\}.$$

如 (i) 所述, 这是不必要的. 其实, 只要 $|\mathcal{D}_k| = k$ 即可.

(iii) 对于 k 值逻辑或混合值逻辑, 其逻辑表达式并不普及应用. 因此, 将 $\mathcal{D}_k$ 用 Δ_k 代替, 将 k 值逻辑或混合值逻辑系统直接表示成代数状态空间表达式是恰当和合理的.

因为 k 值逻辑是一种特殊的混合值逻辑, 故在本节的以下讨论中, 我们只关心混合值逻辑网络, 所有得到的结果均可用于 k 值逻辑网络.

设 (3.3.1) 为一混合值逻辑系统, $X_i(t) \in \mathcal{D}_{k_i}$, $i = 1, 2, \cdots, n$. 记 $x_i(t) = \vec{X}_i$, $x(t) = \ltimes_{i=1}^n x_i(t)$,

$$\kappa := \prod_{i=1}^n k_i.$$

则 (3.3.1) 的分量代数状态空间表达式为

$$\begin{cases} x_1(t+1) = M_1 x(t), \\ x_2(t+1) = M_2 x(t), \\ \qquad \cdots\cdots \\ x_n(t+1) = M_n x(t), \end{cases} \tag{3.3.2}$$

这里 $M_i \in \mathcal{L}_{k_i \times \kappa}$, $i = 1, 2, \cdots, n$.

(3.3.1) 的代数状态空间表达式为

$$x(t+1) = M x(t), \tag{3.3.3}$$

这里

$$M = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{\kappa \times \kappa}.$$

下面考虑混合值逻辑系统的不动点与极限环. 回忆定理 3.1.1 及定理 3.1.2 的证明, 不难发现, 形如 (3.3.3) 的系统, 如果 $M \in \mathcal{L}_{s \times s}$ 是一个逻辑矩阵, 而 $x(t) \in \Delta_s$, 那么, 定理 3.1.1 及定理 3.1.2 的结论对这种系统都成立. 于是有以下结论.

定理 3.3.1　考察混合值逻辑网络 (3.3.1), 设其代数状态空间表达式为 (3.3.3). 那么,

(i) δ_κ^i 为其不动点, 当且仅当, 在 (3.3.3) 中, 系统的结构矩阵 M 的对角元素 $m_{ii}=1$. 因此, (3.3.1) 的不动点个数 N_e 为

$$N_e = \operatorname{tr}(M). \tag{3.3.4}$$

(ii) 记 (3.3.1) 中的长度为 s 的极限环个数为 N_s, 则 N_s 可由下列递推公式算得

$$\begin{cases} N_1 = N_e, \\ N_s = \dfrac{\operatorname{tr}(M^s) - \sum\limits_{k\in\mathcal{P}(s)} kN_k}{s}, \quad 2 \leqslant s \leqslant \kappa. \end{cases} \tag{3.3.5}$$

(iii) 记 $\mathcal{C}_s$ 是长度为 s 的极限环上的点的集合, $\mathcal{D}_a(L)$ 为逻辑矩阵 L 对角元素非零的列的集合, 那么

$$\mathcal{C}_s = \mathcal{D}_a(M^s) \Big\backslash \bigcup_{t\in\mathcal{P}(s)} \mathcal{D}_a(M^t). \tag{3.3.6}$$

构造序列 M^i, $i=1,\cdots,r$, 根据 (3.1.28) 可取上界 $r=\lfloor e^{\kappa/e}\rfloor$.

与布尔网络一样, 我们有如下结论:

(i) 在这个序列中可以找到 r_0 使 M^{r_0} 在序列中重现. 即存在一个 $T>0$, 使得

$$M^{r_0} = M^{r_0+T}. \tag{3.3.7}$$

(ii) 满足 (3.3.7) 的最小的 $r_0>0$ 就是过渡周期 T_t.

(iii) 满足 (3.3.7) 的最小的 $T>0$ 就是所有极限环长度的最小公倍数, 即令 $|C_i|=\ell_i$, $i=1,2,\cdots,s$, 则

$$T = \operatorname{lcm}(\ell_1, \ell_2, \cdots, \ell_s).$$

下面给出一个例子.

例 3.3.1　考察系统

$$\begin{cases} X_1(t+1) = X_2(t)\sigma X_3(t), \\ X_2(t+1) = \varDelta[X_1(t)\wedge X_3(t)], \\ X_3(t+1) = X_2(t)\sigma X_1(t), \end{cases} \tag{3.3.8}$$

这里, $X_1(t), X_3(t) \in \mathcal{D}_2$, $X_2(t) \in \mathcal{D}_3$,

$$\begin{aligned}&\sigma : \mathcal{D}_3 \times \mathcal{D}_2 \to \mathcal{D}_2,\\&\Delta : \mathcal{D}_2 \to \mathcal{D}_3.\end{aligned}$$

其结构矩阵分别为

$$\begin{aligned}&M_\sigma = \delta_2[1,1,2,2,1,2],\\&M_\Delta = \delta_3[3,1].\end{aligned}$$

记 $x_i = \vec{X}_i$, $i = 1,2,3$ 为其向量形式, 则得

$$\begin{aligned}x_1(t+1) &= M_\sigma x_2(t)x_3(t)\\&= M_\sigma(\mathbf{1}_2^{\mathrm{T}} \otimes I_6)x(t) := M_1 x(t),\\x_2(t+1) &= M_\Delta M_c x_1(t)x_3(t)\\&= M_\Delta M_c(I_2 \otimes \mathbf{1}_3^{\mathrm{T}} \otimes I_2)x(t) := M_2 x(t),\\x_3(t+1) &= M_\sigma x_2(t)x_1(t)\\&= M_\sigma W_{[2,3]}(I_6 \otimes \mathbf{1}_2^{\mathrm{T}})x(t) := M_3 x(t).\end{aligned}$$

容易算得

$$\begin{aligned}&M_1 = \delta_2[1,1,2,2,1,2,1,1,2,2,1,2],\\&M_2 = \delta_3[3,1,3,1,3,1,1,1,1,1,1,1],\\&M_3 = \delta_2[1,1,2,2,1,1,1,1,2,2,2,2].\end{aligned}$$

因此,

$$M = M_1 * M_2 * M3 = \delta_{12}[5,1,12,8,5,7,1,1,8,8,2,8].$$

从而可得

$$\begin{aligned}&M^2 = \delta_{12}[5,5,8,1,5,1,5,5,1,1,1,1],\\&M^3 = \delta_{12}[5,5,1,5,5,5,5,5,5,5,5,5],\\&M^k = \delta_{12}[5,5,5,5,5,5,5,5,5,5,5,5], \quad k \geqslant 4.\end{aligned}$$

因此, 混合值网络 (3.3.8) 过渡周期为 4. 它全局收敛于 $x = \delta_{12}^5 \sim (X_1^*, X_2^*, X_3^*)$, 这里

$$\begin{aligned}&x_1^* = (I_2 \otimes \mathbf{1}_6^{\mathrm{T}})\delta_{12}^5 = \delta_2^1,\\&x_2^* = (\mathbf{1}_2^{\mathrm{T}} \otimes I_3 \otimes \mathbf{1}_2^{\mathrm{T}})\delta_{12}^5 = \delta_3^3,\\&x_3^* = (\mathbf{1}_6^{\mathrm{T}} \otimes I_2)\delta_{12}^5 = \delta_2^1.\end{aligned}$$

即 $(X_1^*, X_2^*, X_3^*) = (1,0,1)$.

第 4 章　逻辑系统的能控性

输入输出结构是一个控制系统的基本要素. 从网络结构看, 一个布尔网络, 或者更一般的 k 值或混合值逻辑系统, 会有一些结点, 只有出度没有入度, 这些结点即成 "输入" 或 "控制", 另外一些结点, 只有入度没有出度, 这些结点即成 "输出" 或 "观测". 从实际网络系统看, 例如基因调控网络, 它就有输入输出结构[65]. 因此, 布尔网络或一般逻辑网络的控制问题, 就成为一个具有重要理论意义和应用价值的问题[12,39,40]. 矩阵半张量积及代数状态空间表达式的引入, 大大推进了逻辑系统控制问题的研究[22]. 本章讨论逻辑系统的能控性.

4.1　普通能控性

逻辑系统的普通能控性指一个逻辑系统 (或称逻辑网络) 在自由控制序列 (或称自由输入序列) 下, 从一个状态到另一个状态移动的可实现性. 这是在普通意义下的能控性, 与经典控制理论中的相应概念是一致的.

4.1.1　布尔控制网络的代数状态空间表示

类似于一个布尔网络, 一个布尔控制网络可以用一个网络图来表示. 只是网络图中有输入与输出. 图 4.1.1 是一个布尔控制网络的网络图, 图中有 4 个网络结点 (状态), 2 个输入 (控制), 1 个输出 (观测).

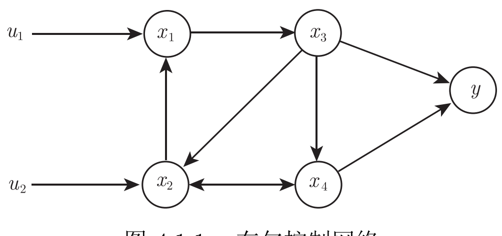


图 4.1.1　布尔控制网络

与布尔网络一样, 网络图只表示结点的连接关系, 结点间具体的逻辑关系则需要一个动态方程来刻画. 一个布尔控制网络可描述如下:

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ \qquad\qquad\cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \end{cases} \tag{4.1.1a}$$

$$Y_j(t) = C_j(X_1(t), X_2(t), \cdots, X_n(t)), \quad j = 1, 2, \cdots, p, \tag{4.1.1b}$$

这里 $X_i(t), U_j(t), Y_k(t) \in \mathcal{D}$, $F_i : \mathcal{D}^{n+m} \to \mathcal{D}$, $h_j : \mathcal{D}^n \to \mathcal{D}$, $i = 1, 2, \cdots, n$, $j = 1, 2, \cdots, m$, $k = 1, 2, \cdots, p$.

例如, 网络图 4.1.1 的一个对应的网络动态系统为

$$\begin{cases} X_1(t+1) = X_2(t) \vee U_1(t), \\ X_2(t+1) = [X_3(t) \wedge X_4(t)] \leftrightarrow U_2(t), \\ X_3(t+1) = \neg X_1(t), \\ X_4(t+1) = X_2(t) \bar{\vee} X_3(t), \end{cases} \tag{4.1.2a}$$

$$Y(t) = X_3(t) \vee (\neg X_4(t)). \tag{4.1.2b}$$

当然, (4.1.2) 只是网络图 4.1.1 所描述的一个可能的网络动态系统. 一个网络图可以对应多个不同的网络动态系统, 所以一个网络图不足以确定一个网络动态系统.

为了使用矩阵半张量积方法, 需要将 (4.1.1) 用它的代数状态空间表示, 其做法与布尔网络类似, 简述如下: 令

$$\begin{array}{lll} x_i(t) = \vec{X}_i(t), & i = 1, 2, \cdots, n, & x(t) = \ltimes_{i=1}^n x_i(t), \\ y_j(t) = \vec{Y}_j(t), & j = 1, 2, \cdots, p, & y(t) = \ltimes_{j=1}^p y_j(t), \\ u_k(t) = \vec{U}_k(t), & k = 1, 2, \cdots, m, & u(t) = \ltimes_{k=1}^m u_k(t), \\ M_i = M_{F_i}, & i = 1, 2, \cdots, n, & \text{为 } F_i \text{ 的结构矩阵}, \\ H_j = M_{C_j}, & j = 1, 2, \cdots, p, & \text{为 } C_j \text{ 的结构矩阵}. \end{array} \tag{4.1.3}$$

记号约定 为叙述简便, 本书其余部分均使用记号 (4.1.3), 即, 作为约定, 总把逻辑网络的状态变量记作 X_i, 输入变量记作 U_i, 输出变量记作 Y_i, 等等, 它们的向量表示分别记作 x_i, u_i, y_i 等.

于是, 可以得到分量代数状态空间表达式

$$\begin{cases} x_1(t+1) = M_1u(t)x(t), \\ x_2(t+1) = M_2u(t)x(t), \\ \qquad\cdots\cdots \\ x_n(t+1) = M_nu(t)x(t), \end{cases} \tag{4.1.4a}$$

$$y_j(t) = H_jx(t), \quad j = 1, 2, \cdots, p. \tag{4.1.4b}$$

进而可得 (整体) 代数状态空间表达式

$$x(t+1) = Mu(t)x(t), \tag{4.1.5a}$$

$$y(t) = Hx(t), \tag{4.1.5b}$$

这里

$$M = M_1 * M_2 * \cdots * M_n,$$
$$H = H_1 * H_2 * \cdots * H_p.$$

例 4.1.1　回忆网络图 4.1.1 和方程 (4.1.2) 所描述的布尔控制系统. 利用记号 (4.1.3), 不难得到分量代数状态空间表达式

$$\begin{cases} x_1(t+1) = M_1u(t)x(t), \\ x_2(t+1) = M_2u(t)x(t), \\ x_3(t+1) = M_3u(t)x(t), \\ x_4(t+1) = M_4u(t)x(t), \end{cases} \tag{4.1.6}$$
$$y_1(t) = H_1x(t),$$

这里

$$\begin{aligned} M_1 = \delta_2[&1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,\\ &1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,\\ &1,1,1,1,2,2,2,2,1,1,1,1,2,2,2,2,\\ &1,1,1,1,2,2,2,2,1,1,1,1,2,2,2,2],\\ M_2 = \delta_2[&1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,\\ &2,1,1,1,2,1,1,1,2,1,1,1,2,1,1,1,\\ &1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,\\ &2,1,1,1,2,1,1,1,2,1,1,1,2,1,1,1],\\ M_3 = \delta_2[&2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1, \end{aligned}$$

$$
\begin{aligned}
&2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1,\\
&2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1,\\
&2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1],\\
M_4 = \delta_2[&2,2,1,1,1,1,2,2,2,2,1,1,1,1,2,2,\\
&2,2,1,1,1,1,2,2,2,2,1,1,1,1,2,2,\\
&2,2,1,1,1,1,2,2,2,2,1,1,1,1,2,2,\\
&2,2,1,1,1,1,2,2,2,2,1,1,1,1,2,2],\\
H_1 = \delta_2[&1,1,2,1,1,1,2,1,1,1,2,1,1,1,2,1].
\end{aligned}
$$

继而可得 (全局) 代数状态空间表达式

$$
\begin{aligned}
&x(t+1) = Mu(t)x(t),\\
&y(t) = Hx(t),
\end{aligned}
\tag{4.1.7}
$$

这里

$$
\begin{aligned}
M &= M_1 * M_2 * M_3 * M_4\\
&= \delta_{16}[4,8,7,7,3,7,8,8,2,6,5,5,1,5,6,6,\\
&\qquad 8,4,3,3,7,3,4,4,6,2,1,1,5,1,2,2,\\
&\qquad 4,8,7,7,11,15,16,16,2,6,5,5,9,13,14,14,\\
&\qquad 8,4,3,3,15,11,12,12,6,2,1,1,13,9,10,10],
\end{aligned}
$$

并且, $H = H_1$.

4.1.2 能控性矩阵与能控能达

下面考虑布尔控制网络的能控性问题, 这时与输出无关.

定义 4.1.1 考察布尔控制网络 (4.1.1a), 其代数状态空间表达式为 (4.1.5a). 设状态 $X^0, X^d \in \mathcal{D}^n$.

(i) 称网络 (4.1.1) 可从 X^0 控制到 X^d, 如果存在 $0 < T < \infty$, 以及一族控制 $U(0), U(1), \cdots, U(T-1)$, 使得网络 (4.1.1) 在该控制下, 其轨线能从 $X(0) = X^0$ 到达 $X(T) = X^d$.

(ii) 称网络 (4.1.1) 在 X^0 能控, 如果对任何 $X \in \mathcal{D}^n$, X^0 到 X 均能控.

(iii) 称网络 (4.1.1) 到 X^d 能达, 如果对任何 $X \in \mathcal{D}^n$, X 到 X^d 均能控.

(iv) 称网络 (4.1.1) (完全) 能控, 如果对任何 $X^1, X^2 \in \mathcal{D}^n$, X^1 到 X^2 均能控.

记

$$\begin{aligned} & X^0 = (X_1^0, X_2^0, \cdots, X_n^0), \\ & X^d = (X_1^d, X_2^d, \cdots, X_n^d), \\ & x^0 = \vec{X}^0 = \delta_{2^n}^j, \\ & x^d = \vec{X}^d = \delta_{2^n}^i. \end{aligned} \tag{4.1.8}$$

注意, 网络 (4.1.1) 与其代数状态空间表达式 (4.1.5) 是一一对应的, 因此, 讨论 (4.1.1) 的能控性与 (4.1.5) 的能控性是一样的. 我们的目的, 就是通过 (4.1.5) 将 (4.1.1) 的能控性转化为一个代数问题.

考察 (4.1.5a), 在能控性问题中我们使用布尔矩阵的运算, 定义如下.

定义 4.1.2 [9]　(i) 设 $a, b \in \mathcal{D}$, 则

$$\begin{aligned} & a +_{\mathcal{B}} b = a \vee b, \\ & a \times_{\mathcal{B}} b = a \wedge b. \end{aligned} \tag{4.1.9}$$

(ii) 设 $A \in \mathcal{B}_{m\times s}$, $B \in \mathcal{B}_{s\times n}$, 则

$$A \times_{\mathcal{B}} B = C \in \mathcal{B}_{m\times n}, \tag{4.1.10}$$

这里, (记 $A = (a_{i,j})$, $B = (b_{i,j})$, $C = (c_{i,j})$)

$$c_{i,j} = \sum_{k=1}^{s}{}_{\mathcal{B}} a_{i,k} \times_{\mathcal{B}} b_{k,j}.$$

(iii)

$$A^{(k)} = \underbrace{A \times_{\mathcal{B}} A \times_{\mathcal{B}} \cdots \times_{\mathcal{B}} A}_{k}. \tag{4.1.11}$$

定义 4.1.3　设 $A \in \mathcal{B}_{m\times n}$, $B \in \mathcal{B}_{p\times q}$, $t = \operatorname{lcm}(n, p)$, 则 A 与 B 的布尔矩阵半张量积定义为

$$A \ltimes_{\mathcal{B}} B := \left(A \otimes I_{t/n}\right) \times_{\mathcal{B}} \left(B \otimes I_{t/p}\right). \tag{4.1.12}$$

注 4.1.1　严格地说, 定义 4.1.3 是定义了 MM-1 型布尔矩阵半张量积. 当然, 其他类型的布尔矩阵半张量积可类似定义.

回到 (4.1.5a), 定义

$$T := \sum_{i=1}^{m}{}_{\mathcal{B}} M \delta_m^i. \tag{4.1.13}$$

称 T 为网络 (4.1.1) 的状态转移矩阵.

下面这个引理直接来自定义, 它说明状态转移矩阵的物理意义.

引理 4.1.1 记 $T=(t_{ij})$, $T^{(s)}=(t^s_{i,j})$, $s>1$. (i) 存在控制 $u\in\Delta_{2^m}$ 使网络从 $x^0=\delta^j_{2^n}$ 一步到达 $x^d=\delta^i_{2^n}$, 即

$$\delta^i_{2^n}=Mu\delta^j_{2^n}, \tag{4.1.14}$$

当且仅当,

$$t_{ij}=1.$$

(ii) 存在控制 $u^0,u^1,\cdots,u^{k-1}\in\Delta_{2^m}$ 使网络从 $x^0=\delta^j_{2^n}$ k 步到达 $x^d=\delta^i_{2^n}$, 即

$$\begin{aligned}&x(i+1)=Mu^ix(i),\quad i=0,1,\cdots,k-1,\\&x(0)=\delta^j_{2^n},\quad x(k)=\delta^i_{2^n},\end{aligned} \tag{4.1.15}$$

当且仅当,

$$t^k_{ij}=1.$$

下面这个引理对讨论布尔网络能控性是必要的.

引理 4.1.2 设 $T\in\mathcal{B}_{k\times k}$, 那么

$$\sum_{i=1}^{\infty}{}_{\mathcal{B}}T^{(i)}=\sum_{i=1}^{k}{}_{\mathcal{B}}T^{(i)}. \tag{4.1.16}$$

证明 设 $x^0=\delta^j_k$. 定义从 x^0 出发, 依 T 一步能达集合为

$$R_1(x^0):=\{\delta^t_k\mid\delta^t_k\leqslant Tx^0\},$$

s 步能达集合递推地定义为

$$R_s(x^0):=R_{s-1}(x^0)\cup\left\{d^t_k\mid\delta^t_k\leqslant T^{(s)}x^0\right\}.$$

则显然 $R_1\subset R_2\subset R_3\subset\cdots$ 是一个非降的集合序列. 如果

$$|R_1|<|R_2|<\cdots<|R_k|,$$

则有 $|R_k|=k$, 于是

$$t^{\ell}_{i,j}=1,\quad \ell>k. \tag{4.1.17}$$

如果存在 $1\leqslant n<k$, 使得 $R_n=R_{n+1}$. 不妨设 R_n 为第一个 (最小的 n) 满足等式条件的集合, 那么显然,

$$R_m=R_n,\quad m>n.$$

这时,

$$t_{i,j}^{\ell}=\begin{cases}1, & \delta_k^i\in R_n,\\ 0, & \delta_k^i\notin R_n, \quad \ell>k.\end{cases} \tag{4.1.18}$$

由 (4.1.17) 及 (4.1.18) 可得

$$t_{i,j}^{\ell}=t_{i,j}+_{\mathcal{B}}t_{i,j}^2+_{\mathcal{B}}\cdots+_{\mathcal{B}}t_{i,j}^k, \quad \ell>k. \tag{4.1.19}$$

由于 $1\leqslant i,j\leqslant k$ 是任意的, (4.1.18) 蕴涵了 (4.1.16). □

利用 (4.1.13) 所定义的 T, 我们定义如下能控性矩阵

$$\mathcal{C}:=\sum_{i=1}^{2^n}{}_{\mathcal{B}}T^{(i)}. \tag{4.1.20}$$

根据引理 4.1.1 及引理 4.1.2 不难推得以下结论.

定理 4.1.1 考察布尔控制网络 (4.1.1a), 其代数状态空间表达式为 (4.1.5a). 设状态 $X^0,X^d\in\mathcal{D}^n$. $x^0=\vec{X}^0=\delta_{2^n}^j$, $x^d=\vec{X}^d=\delta_{2^n}^i$, 能控性矩阵 $\mathcal{C}=(c_{i,j})$.

(i) 网络 (4.1.1) 可从 X^0 控制到 X^d, 当且仅当,

$$c_{i,j}=1.$$

(ii) 网络 (4.1.1) 在 X^0 可控, 当且仅当,

$$\mathrm{Col}_j(\mathcal{C})=\mathbf{1}_{2^n}.$$

(iii) 网络 (4.1.1) 到 X^d 可达, 当且仅当,

$$\mathrm{Row}_i(\mathcal{C})=\mathbf{1}_{2^n}^{\mathrm{T}}.$$

(iv) 网络 (4.1.1) 可控, 当且仅当,

$$\mathcal{C}=\mathbf{1}_{2^n\times 2^n}.$$

例 4.1.2 回忆例 4.1.1, 利用 (4.1.7) 中的 M, 直接算得

$$\begin{aligned}T=&M\delta_4^1+_{\mathcal{B}}M\delta_4^2+_{\mathcal{B}}M\delta_4^3+_{\mathcal{B}}M\delta_4^4\\ =&\big[\delta_{16}^4+\delta_{16}^8,\delta_{16}^4+\delta_{16}^8,\delta_{16}^2+\delta_{16}^7,\delta_{16}^2+\delta_{16}^7,\delta_{16}^2+\delta_{16}^7+\delta_{16}^{11}+\delta_{16}^{15},\\ &\delta_{16}^2+\delta_{16}^7+\delta_{16}^{11}+\delta_{16}^{15},\delta_{16}^4+\delta_{16}^8+\delta_{16}^{12}+\delta_{16}^{16},\delta_{16}^4+\delta_{16}^8+\delta_{16}^{12}+\delta_{16}^{16},\\ &\delta_{16}^2+\delta_{16}^6,\delta_{16}^2+\delta_{16}^6,\delta_{16}^1+\delta_{16}^5,\delta_{16}^1+\delta_{16}^5,\delta_{16}^1+\delta_{16}^5+\delta_{16}^9+\delta_{16}^{13},\\ &\delta_{16}^1+\delta_{16}^5+\delta_{16}^9+\delta_{16}^{13},\delta_{16}^2+\delta_{16}^6+\delta_{16}^{10}+\delta_{16}^{14},\delta_{16}^2+\delta_{16}^6+\delta_{16}^{10}+\delta_{16}^{14}\big].\end{aligned}$$

进而可得

$$\mathcal{C} = \sum_{i=1}^{16} {}_{\mathcal{B}}T^{(i)} = \mathbf{1}_{16\times 16}.$$

于是可知, (4.1.1a) 完全可控.

一个布尔网络的能控性可以形象地用网络的状态转移图来刻画. 状态转移图的结点是状态, 每条边注明状态转移及相应的控制. 只是当结点不是很少时, 这个图很复杂. 下面用一个例子来说明.

例 4.1.3 考察如下网络

$$\begin{cases} X_1(t+1) = X_2(t) \vee U(t), \\ X_2(t) = X_1(t) \wedge U(t). \end{cases} \tag{4.1.21}$$

不难直接画出它的状态转移图, 见图 4.1.2. 从图中不难看出, 从任何一点出发, 都可到达 $(1,1)$, $(1,0)$ 或 $(0,0)$, 但都不能到达 $(0,1)$. 不难算出, 这个网络的能控性矩阵为

$$\mathcal{C} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

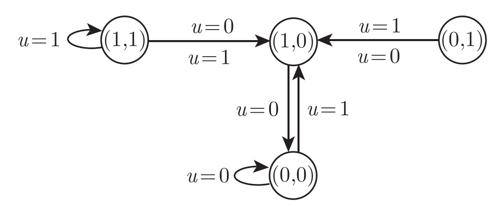


图 4.1.2 网络 (4.1.21) 的状态转移图

根据定理 4.1.1, 我们可以得到同样的结论.

4.1.3 一般逻辑系统的能控性

定义 4.1.4 考察网络方程 (4.1.1),

(i) 如果 $X_i(t) \in \mathcal{D}_{r_i}$, $U_\alpha \in \mathcal{D}_{\ell_\alpha}$, $Y_j \in \mathcal{D}_{s_j}$, $i = 1, 2, \cdots, n$, $\alpha = 1, 2, \cdots, m$, $j = 1, 2, \cdots, p$. 记

$$r = \prod_{i=1}^{n} r_i, \quad \ell = \prod_{\alpha=1}^{m} \ell_\alpha, \quad s = \prod_{j=1}^{p} s_j,$$

$F_i:\mathcal{D}_{r\ell}\to\mathcal{D}_{r_i}$, $C_j:\mathcal{D}_r\to\mathcal{D}_{s_j}$ 为逻辑函数, 那么, (4.1.1) 称为一个混合值控制逻辑网络.

(ii) (4.1.1) 称为一个 k 值逻辑控制网络, 如果

$$r_i=\ell_\alpha=s_j=k\geqslant 3,\quad i=1,\cdots,n,\quad \alpha=1,\cdots,m,\quad j=1,\cdots,p.$$

记 $x_i=\vec{X}_i$, $u_\alpha=\vec{U}_\alpha$, $y_j=\vec{Y}_j$, $x=\ltimes_{i=1}^n x_i$, $u=\ltimes_{\alpha=1}^m u_\alpha$, $y=\ltimes_{j=1}^p y_j$, 则可得到形如 (4.1.5) 混合值逻辑网络的代数状态空间表达式, 现在

$$M\in\mathcal{L}_{k\times k\ell},\quad H\in\mathcal{L}_{s\times k}. \tag{4.1.22}$$

定义状态转移矩阵

$$T=\sum_{\alpha=1}^{\ell}{}_{\mathcal{B}}M\delta_\ell^\alpha\in\mathcal{L}_{k\times k}. \tag{4.1.23}$$

再定义能控性矩阵

$$\mathcal{C}=\sum_{i=1}^{k}{}_{\mathcal{B}}T^{(i)}. \tag{4.1.24}$$

那么, 我们不难得到与布尔控制网络一样的能控性结论.

推论 4.1.1　对于混合值逻辑控制网络 (4.1.1), 设其能控性矩阵由 (4.1.24) 定义, 则定理 4.1.1 成立.

下面给出一个例子.

例 4.1.4　考察混合值逻辑控制系统

$$\begin{cases}X_1(t+1)=U(t)\Box X_2(t),\\ X_2(t+1)=U(t)\triangle X_1(t),\end{cases} \tag{4.1.25}$$

这里 $X_1(t),U(t)\in\mathcal{D}_3$, $X_2(t)\in\mathcal{D}_2$, $\Box:\mathcal{D}_3\times\mathcal{D}_2\to\mathcal{D}_3$ 的结构矩阵为

$$M_\Box=\delta_3[1,3,2,2,3,1],$$

$\triangle:\mathcal{D}_3\times\mathcal{D}_3\to\mathcal{D}_2$ 的结构矩阵为

$$M_\triangle=\delta_2[1,2,2,1,2,1,2,1,1].$$

不难算出, (4.1.25) 的分量代数状态空间表达式为

$$x_1(t+1)=M_\Box u(t)x_2(t)$$

$$
\begin{aligned}
&= M_{\square}\left(I_3 \otimes \mathbf{1}_3^{\mathrm{T}} \otimes I_2\right) u(t)x(t) \\
&:= M_1 u(t)x(t),
\end{aligned} \tag{4.1.26}
$$

这里

$$M_1 = \delta_3[1,3,1,3,1,3,2,2,2,2,2,2,3,1,3,1,3,1].$$

$$
\begin{aligned}
x_2(t+1) &= M_{\triangle} u(t) x_1(t) \\
&= M_{\triangle}(I_9 \otimes \mathbf{1}_2^{\mathrm{T}}) u(t)x(t) \\
&:= M_2 u(t)x(t),
\end{aligned} \tag{4.1.27}
$$

其中

$$M_2 = \delta_3[1,1,2,2,2,2,1,1,2,2,1,1,2,2,1,1,1,1].$$

于是

$$M = \delta_6[1,5,2,6,2,6,2,3,4,4,3,3,6,2,5,1,5,1]. \tag{4.1.28}$$

进而可得状态转移矩阵为

$$T = M\delta_3^1 +_{\mathcal{B}} M\delta_3^2 +_{\mathcal{B}} M\delta_3^3 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}. \tag{4.1.29}$$

最后可得, 能控性矩阵为

$$\mathcal{C} = \sum_{i=1}^{6} {}_{\mathcal{B}} T^{(i)} = \mathbf{1}_{6\times 6}, \tag{4.1.30}$$

因此, 混合值逻辑控制网络 (4.1.25) 是完全能控的.

4.2 集合能控性

与普通能控性不同, 集合能控性研究系统能否从一个集合到达另一个集合. 除了本身的重要性外, 集合能控性在逻辑系统的控制理论中还起着一种工具性的作用, 因为许多其他控制问题可转化为集合能控性问题, 从而得到解决.

4.2.1　集合能控的充要条件

首先给出集合能控的严格定义, 为此, 我们必须先对系统出发的子集和到达的子集做一说明.

考察布尔控制网络 (4.1.1), 其结点集合为 $N=\{1,2,\cdots,n\}$, 状态集合为 2^N, 设 P^0 和 P^d 为两个状态子集的集合, 记作

$$P^0:=\left\{s_1^0,s_2^0,\cdots,s_\alpha^0\right\},\quad P^d:=\left\{s_1^d,s_2^d,\cdots,s_\beta^d\right\}, \tag{4.2.1}$$

这里 $s_i^0\subset 2^N$, $i=1,2,\cdots,\alpha$, $s_j^d\subset 2^N$, $j=1,2,\cdots,\beta$ 均为状态集合的子集. P^0 称为初始集合集, P^d 称为目标集合集.

利用初始集合集与目标集合集, 集合能控性可定义如下.

定义 4.2.1　考察布尔控制网络 (4.1.1a) 以及由 (4.2.1) 所给定的初始集合集与目标集合集.

(i) 系统 (4.1.1) 称为从 $s_j^0\in P^0$ 到 $s_i^d\in P^d$ 集合能控, 如果存在 $x^0\in s_j^0$ 及 $x^d\in s_i^d$, 使得系统 (4.1.1) 可从状态 x^0 控制到状态 x^d.

(ii) 系统 (4.1.1) 称为在 $s_j^0\in P^0$ 集合能控, 如果对任何 $s_i^d\in P^d$ 系统 (4.1.1) 从集合 s_j^0 到集合 s_i^d 均集合能控.

(iii) 系统 (4.1.1) 称为到 $s_i^d\in P^d$ 集合能达, 如果对任何 $s_j^0\in P^0$ 系统 (4.1.1) 从集合 s_j^0 到集合 s_i^d 均集合能控.

(iv) 系统 (4.1.1) 称为从 P^0 到 P^d 完全集合能控, 如果系统 (4.1.1) 从任何 $s_j^0\in P^0$ 到任何 $s_i^d\in P^d$ 均集合能控.

注意, 在这个定义里, 我们并不要求初始集 (或目标集) 中的各子集互不相关, 这会为应用带来很大方便. 图 4.2.1 给出一个清晰的刻画.

为了研究集合能控性, 我们需要集合的向量或矩阵刻画.

定义 4.2.2　设 $W=\{w_1,w_2,\cdots,w_n\}$ 为一有限集.

(i) $s\subset W$, s 的示性向量, 记作 $V(s)\in\mathbb{R}^n$, 定义如下:

$$(V(s))_i=\begin{cases}1, & w_i\in s,\\ 0, & w_i\notin s.\end{cases} \tag{4.2.2}$$

(ii) 设 $P=\{s_1,s_2,\cdots,s_r\}\subset 2^W$, 即 $s_i\subset W$, $i=1,2,\cdots,r$. P 的示性矩阵, 记作 $J(P)\in\mathcal{B}_{n\times r}$, 定义为

$$J(P)=\left[V(s_1),V(s_2),\cdots,V(s_r)\right]. \tag{4.2.3}$$

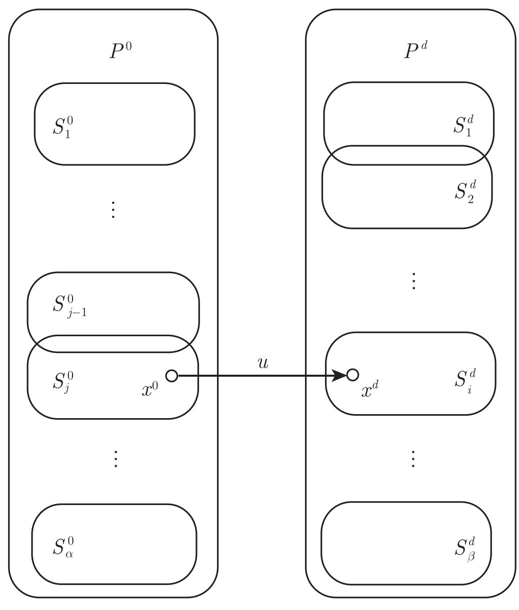


图 4.2.1 集合能控性

考察由 (4.2.1) 所给定的初始集合集与目标集合集. 记

$$J(P^0) := J_0, \quad J(P^d) = J^d.$$

然后集合能控性矩阵, 记作 $\mathcal{C}_S$, 定义如下:

$$\mathcal{C}_S := J_d^{\mathrm{T}} \times_{\mathcal{B}} \mathcal{C} \times_{\mathcal{B}} J_0 \in \mathcal{B}_{\beta \times \alpha}, \tag{4.2.4}$$

这里, $\mathcal{C}$ 是原系统的能控性矩阵.

类似于能控性与能控性矩阵的关系, 不难证明集合能控性与集合能控性矩阵有如下关系.

定理 4.2.1 考察布尔控制网络 (4.1.1a) 以及由 (4.2.1) 所给定的初始集合集与目标集合集. 其相应的集合能控性矩阵 $\mathcal{C}_S = (c_{i,j})$ 由 (5.2.4) 定义. 那么,

(i) 布尔控制网络 (4.1.1a) 从 s_j^0 到 s_i^d 集合能控, 当且仅当, $c_{i,j} = 1$;

(ii) 布尔控制网络 (4.1.1a) 在 s_j^0 集合能控, 当且仅当, $\mathrm{Col}_j(\mathcal{C}_S) = \mathbf{1}_\beta$;

(iii) 布尔控制网络 (4.1.1a) 到 s_i^d 集合能控, 当且仅当, $\mathrm{Row}_i(\mathcal{C}_S) = \mathbf{1}_\alpha^{\mathrm{T}}$;

(iv) 布尔控制网络 (4.1.1a) 从 P^0 到 P^d 完全集合能控, 当且仅当, $\mathcal{C}_S = \mathbf{1}_{\alpha \times \beta}$.

下面给出一个算例.

例 4.2.1 考察下列布尔控制网络[22]:

$$\begin{cases} X_1(t+1) = (X_1(t) \leftrightarrow X_2(t)) \vee U_1(t), \\ X_2(t+1) = \neg X_1(t) \wedge U_2(t). \end{cases} \tag{4.2.5}$$

不难算出[22]

$$\mathcal{C}=\begin{bmatrix}1&1&1&1\\1&1&1&1\\0&0&1&0\\1&1&1&1\end{bmatrix}.$$

(i) 设

$$\begin{cases}P^d=\left\{s_1^d=(\delta_4^1,\delta_4^2),\quad s_2^d=(\delta_4^3,\delta_4^4)\right\},\\P^0=\{s_1^0=(\delta_4^1),\quad s_2^0=(\delta_4^2,\delta_4^3,\delta_4^4)\},\end{cases}\tag{4.2.6}$$

则有

$$J_d=\begin{bmatrix}1&0\\1&0\\0&1\\0&1\end{bmatrix},\quad J_0=\begin{bmatrix}1&0\\0&1\\0&1\\0&1\end{bmatrix},$$

于是可得

$$\mathcal{C}_S=J_d^{\mathrm{T}}\mathcal{C}J_0=\begin{bmatrix}1&1\\1&1\end{bmatrix}.$$

因此, 网络 (5.2.5) 关于由 (4.2.6) 所定义的初始集集合 P^0 和目标集集合 P^d 是集合完全能控的.

(ii) 设

$$\begin{cases}P^d=\left\{s_1^d=(\delta_4^3)\right\},\\P^0=\{s_1^0=(\delta_4^1,\delta_4^2,\delta_4^3),s_2^0=(\delta_4^1,\delta_4^4)\},\end{cases}\tag{4.2.7}$$

那么

$$J_d=\begin{bmatrix}0\\0\\1\\0\end{bmatrix},\quad J_0=\begin{bmatrix}1&1\\1&0\\1&0\\0&1\end{bmatrix},$$

于是

$$\mathcal{C}_S=J_d^{\mathrm{T}}\mathcal{C}J_0=[1,0].$$

因此, 网络 (5.2.5) 关于由 (4.2.7) 所定义的初始集集合 P^0 和目标集集合 P^d 不是集合完全能控的. 但它在 s_1^0 能控.

4.2.2 网络输入及混合输入的能控性

定义 4.2.3 考察布尔控制系统 (4.1.1), 如果其控制满足另一个网络方程, 称为控制网络, 如下:

$$\begin{cases} U_1(t+1) = G_1(U_1(t), \cdots, U_m(t)), \\ U_2(t+1) = G_2(U_1(t), \cdots, U_m(t)), \\ \qquad \cdots\cdots \\ U_m(t+1) = G_m(U_1(t), \cdots, U_m(t)). \end{cases} \tag{4.2.8}$$

则称 (4.1.1)—(4.2.8) 为网络输入布尔控制系统.

对于网络输入布尔控制系统, 真正的控制量只有控制系统的初值 $U_\alpha(0)$, $\alpha = 1, 2, \cdots, m$. 下面考虑混合输入的布尔控制网络.

定义 4.2.4 考察布尔控制系统

$$\begin{cases} X_1(t+1) = F_1(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_p(t), V_1(t), \cdots, V_q(t)), \\ X_2(t+1) = F_2(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_p(t), V_1(t), \cdots, V_q(t)), \\ \qquad \cdots\cdots \\ X(t+1) = F_1(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_p(t), V_1(t), \cdots, V_q(t)), \end{cases} \tag{4.2.9}$$

这里, $U_i(t)$, $i = 1, \cdots, p$ 产生于控制网络 (4.2.8) $(m = p)$, $V_j(t)$, $j = 1, \cdots, q$ 为普通输入 (即自由序列输入), 则 (4.2.9) 称为混合输入的布尔控制网络.

因为网络输入的布尔控制网络可以看作混合输入的布尔控制网络的特殊情况, 我们直接讨论混合输入的布尔控制网络的能控性. 首先将 (4.2.9) 及 (4.2.8) 分别写成代数状态空间表达式如下:

$$x(t+1) = Lv(t)u(t)x(t), \tag{4.2.10}$$

以及

$$u(t+1) = Gu(t), \tag{4.2.11}$$

这里 $x(t) = \ltimes_{i=1}^n x_i(t)$, $u(t) = \ltimes_{j=1}^p u_j(t)$, $v(t) = \ltimes_{k=1}^q v_k(t)$, $L \in \mathcal{L}_{2^n \times 2^{p+q}}$, $G \in \mathcal{L}_{2^p \times 2^p}$.

将 (4.2.10) 及 (4.2.11) 放到一起, 则有

$$\begin{cases} u(t+1) = Gu(t), \\ x(t+1) = Lv(t)u(t)x(t). \end{cases} \tag{4.2.12}$$

令新状态变量为 $w(t) = u(t)x(t)$, 那么 (4.2.12) 变为

$$w(t+1) = \Phi v(t)w(t), \tag{4.2.13}$$

这里

$$\Phi = \left[G(\mathbf{1}_{2^q}^{\mathrm{T}} \otimes I_{2^p} \otimes \mathbf{1}_{2^n}^{\mathrm{T}})\right] * L,$$

其中 $*$ 是 Khatri-Rao 积. 对系统 (4.2.13) 定义两个子集族

$$P^d = P^0 = \{s_i \mid i = 1, \cdots, 2^n\}, \tag{4.2.14}$$

这里

$$s_i = \left\{w \mid \mathbf{1}_{2^r}^{\mathrm{T}} w = \delta_{2^n}^i\right\}.$$

利用这组初始集与目标集, 下面的定理说明, 混合输入布尔网络的能控性问题可转化为集合能控性问题.

定理 4.2.2 考察混合输入布尔控制网络 (4.2.9), 其中 $U_i, i = 1, 2, \cdots, p$ 为网络输入, 满足控制网络 (4.2.8), $V_i, i = 1, 2, \cdots, q$ 为自由输入. 那么, (4.2.9) 在混合输入下完全能控, 当且仅当, 辅助系统 (4.2.13) 关于由 (4.2.14) 定义的初始集与目标集完全集合能控.

证明 (充分性) 即 (4.2.13) 关于 (4.2.14) 集合能控 $\Rightarrow$ (4.2.9) 关于网络输入 (4.2.8) 混合能控.

设 $w_1 = u_1x_1$, $w_2 = u_2x_2$. 由定义式 (4.2.14) 可知

$$w_1,\ w_2 \in s_i \Leftrightarrow x_1 = x_2.$$

也就是说, 每个 s_i 对应一个唯一的 x_i. 因此, s_j 到 s_i 集合能控表示存在一族 $v(t)$, $t = 0, 1, \cdots, T-1$, 使相应 $s(t)$ 从 $s(0) = s_j$ 到达 $s(T) = s_i$, 记 $s(0) = u(0)x_j$, 那么, 混合输入 $u(0)$ 与 $v(t)$, $t = 0, 1, \cdots, T-1$ 可使 (4.2.9) 从 x_j 到达 x_i.

(必要性) 设 (4.2.8) 混合能控. 任给 s_j 与 s_i, 设它们分别对应 x_j 与 x_i, 设计控制 $u(0)$ 与 $v(t)$, $t = 0, 1, \cdots, T-1$, 使 (4.2.8) 从 x_j 到达 x_i, 那么, 根据集合能控性的定义, 这族 $v(t)$, $t = 0, 1, \cdots, T-1$ 也让 s_j 控制到 s_i 了. □

注 4.2.1 不难算出, (4.2.14) 所定义的初始集与目标集的示性矩阵为

$$J_0 = J_d := J = [\underbrace{I_{2^n}, I_{2^n}, \cdots, I_{2^n}}_{2^p}]^{\mathrm{T}}. \tag{4.2.15}$$

据此可构造 (4.2.13) 的集合能控性矩阵.

$$\mathcal{C}_S = J^{\mathrm{T}} \mathcal{C} J.$$

用它即可检验 (4.2.13) 的集合能控性.

由于 s_i 与 x_i 的一一对应, 可以得到如下更详尽的结果.

推论 4.2.1 记辅助系统 (4.2.13) 关于由 (4.2.14) 定义的初始集与目标集的集合能控性矩阵为 $\mathcal{C}_S = (c_{i,j})$. 那么, (4.2.9) 在混合输入下的能控性满足以下性质:

(i) 网络 (4.2.9) 可从 X^0 控制到 X^d, 当且仅当,

$$c_{i,j} = 1.$$

(ii) 网络 (4.2.9) 在 X^0 可控, 当且仅当,

$$\mathrm{Col}_j(\mathcal{C}_S) = \mathbf{1}_{2^n}.$$

(iii) 网络 (4.2.9) 到 X^d 可达, 当且仅当,

$$\mathrm{Row}_i(\mathcal{C}_S) = \mathbf{1}_{2^n}^{\mathrm{T}}.$$

(iv) 网络 (4.2.9) 完全可控, 当且仅当,

$$\mathcal{C}_S = \mathbf{1}_{2^n \times 2^n}.$$

下面讨论一个例子.

例 4.2.2 考察一个布尔控制网络

$$\begin{cases} X_1(t+1) = (U_1(t) \vee V(t)) \wedge X_2(t), \\ X_2(t+1) = U_2(t) \leftrightarrow X_3(t), \\ X_3(t+1) = (V(t) \bar{\vee} X_1(t)) \vee X_2(t), \end{cases} \tag{4.2.16}$$

这里, $V(t)$ 是自由输入序列, $\{U_1(t), U_2(t)\}$ 是网络化输入, 其输入网络为

$$\begin{cases} U_1(t+1) = U_1(t) \wedge U_2(t), \\ U_2(t+1) = \neg U_2(t). \end{cases} \tag{4.2.17}$$

令

$$w(t) = u(t)x(t),$$

直接计算可得, 关于 w 的辅助系统的代数状态空间表示为

$$w(t+1) = \Phi v(t) w(t), \tag{4.2.18}$$

这里,

$$\begin{aligned}\Phi = \delta_{32}[&9,11,14,16,9,11,13,15,19,17,24,22,19,\\&17,23,21,25,27,30,32,25,27,29,31,19,17,\\&24,22,19,17,23,21,9,11,13,15,9,11,14,16,\\&19,17,23,21,19,17,24,22,29,31,29,31,\\&29,31,30,32,23,21,23,21,23,21,24,22]\\:= [&\Phi_1,\Phi_2].\end{aligned}$$

则其状态转移阵为

$$T = \Phi_1 +_{\mathcal{B}} \Phi_2.$$

由此可得能控性矩阵为

$$\mathcal{C} = \sum_{j=1}^{2^{n+p}} {}_{\mathcal{B}} T^{(j)}.$$

根据 (4.2.15) 可得示性矩阵

$$J = [\underbrace{I_8,\cdots,I_8}_{4}]^{\mathrm{T}}.$$

于是, 集合能控性矩阵为

$$\begin{aligned}\mathcal{C}_S &= J^{\mathrm{T}}\mathcal{C}J\\&= J^{\mathrm{T}}\left(\sum_{i=1}^{32} {}_{\mathcal{B}} T^{(i)}\right)J\\&= \begin{bmatrix}1&1&1&1&1&1&1&1\\0&0&0&0&0&0&0&0\\1&1&1&1&1&1&1&1\\0&0&0&0&0&0&0&0\\1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1\\0&1&1&1&0&1&1&1\end{bmatrix}.\end{aligned}$$

于是可知, $\delta_8^1 \sim (1,1,1)$, $\delta_8^3 \sim (1,0,1)$, $\delta_8^5 \sim (0,1,1)$, $\delta_8^6 \sim (0,1,0)$, $\delta_8^7 \sim (0,0,1)$ 是可达的; $\delta_8^2 \sim (1,1,0)$ 及 $\delta_8^4 \sim (1,0,0)$ 完全不可达. $\delta_8^8 \sim (0,0,0)$ 也是不可达的. 具体地说, 从 $\delta_8^1 \sim (1,1,1)$ 或 $\delta_8^5 \sim (0,1,1)$ 出发, 则达不到 $(0,0,0)$, 但从其他点出发能达到 $(0,0,0)$.

因此, 网络 (4.2.16) 不是混合控制下完全能控的.

下面考虑网络输入的能控性. 目的是将混合输入能控性的结果应用到这一种特殊情况中去.

考察一个自治的布尔网络

$$x(t+1) = Mx(t). \tag{4.2.19}$$

类似于自由输入的布尔网络, 我们可以形式地构造它的"能控性矩阵"如下:

$$\mathcal{C}_0 := \sum_{i=1}^{2^n}{}_{\mathcal{B}} M^{(i)}. \tag{4.2.20}$$

记 $\mathcal{C}_0 = (c^0_{i,j})$. 不难看出, 类似于 $\mathcal{C}$, $\mathcal{C}_0$ 的元素 $c^0_{i,j} = 1$ 表明网络 (4.2.19) 的状态 $\delta^j_{2^n}$ 经若干有限时间 $T \leqslant 2^n$ 可到达 $\delta^i_{2^n}$. 因此, $\mathcal{C}_0$ 起着类似 $\mathcal{C}$ 的作用.

下面考虑网络化输入系统

$$\begin{cases} X_1(t+1) = F_1(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_p(t)), \\ X_2(t+1) = F_2(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_p(t)), \\ \qquad \cdots\cdots \\ X(t+1) = F_1(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_p(t)), \end{cases} \tag{4.2.21}$$

这里, $U_i(t)$, $i = 1, \cdots, p$ 产生于控制网络 (4.2.8).

则辅助系统相应的代数状态空间表达式为

$$\begin{cases} u(t+1) = Gu(t), \\ x(t+1) = Lu(t)x(t). \end{cases} \tag{4.2.22}$$

类似于混合输入, 我们令 $w(t) = u(t)x(t)$, 然后将 (4.2.22) 转化为

$$w(t+1) = \Phi w(t), \tag{4.2.23}$$

这里, $\Phi = [G(I_{2^r} \otimes \mathbf{1}^{\mathrm{T}}_{2^n})] * L$. 然后构造

$$\mathcal{C}_0 = \sum_{j=1}^{2^{n+p}}{}_{\mathcal{B}} \Psi^{(j)}, \tag{4.2.24}$$

这里,

$$\Psi := \sum_{j=1}^{2^p}{}_{\mathcal{B}} \Phi \delta^j_{2^p}. \tag{4.2.25}$$

于是有 (4.2.22) 的集合能控性矩阵

$$\mathcal{C}_S^0 := J^{\mathrm{T}}\mathcal{C}_0 J, \tag{4.2.26}$$

这里, J 由 (4.2.15) 确定. 于是我们有如下推论.

推论 4.2.2　网络化输入系统 (4.2.21) 完全能控, 当且仅当,

$$\mathcal{C}_S^0 > 0. \tag{4.2.27}$$

4.3　牵 制 控 制

对于一个复杂的布尔网络, 例如基因调控网络, 常常会有这种情况, 即只有一部分结点允许加控制, 而其余结点是不允许 (或无法) 加控制的. 这些容许加控制的结点称为牵制结点 (pinning node), 只在牵制结点加的控制称牵制控制 (pinning control).

4.3.1　牵制控制网络的代数状态空间表示

考察布尔网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), \cdots, X_n(t)), \\ X_2(t+1) = F_2(X_1(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), \cdots, X_n(t)). \end{cases} \tag{4.3.1}$$

设 $N_p = \{X_1, X_2, \cdots, X_m\}$ 为牵制结点集合, $1 \leqslant m < n$. 那么, (4.3.1) 在牵引控制下变为

$$\begin{cases} X_1(t+1) = F_1(X_1(t), \cdots, X_n(t))\Diamond_1 U_1(t), \\ X_2(t+1) = F_2(X_1(t), \cdots, X_n(t))\Diamond_2 U_2(t), \\ \qquad\qquad \cdots\cdots \\ X_m(t+1) = F_m(X_1(t), \cdots, X_n(t))\Diamond_r U_m(t), \\ X_{m+1}(t+1) = F_{m+1}(X_1(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), \cdots, X_n(t)), \end{cases} \tag{4.3.2}$$

这里, U_α, $\alpha = 1, 2, \cdots, m$ 称为牵引控制, $\Diamond_\alpha$ 是二元布尔算子, 称为牵引控制的注入模式 (injection mode), (4.3.2) 称为牵引控制布尔网络.

注 4.3.1 为了讨论的方便, 我们只关心形如 (4.3.2) 的牵引控制布尔网络. 这里作了如下简化.

(i) 设牵引结点集合为 $N_p = \{X_1, X_2, \cdots, X_m\}$. 这不失一般性, 因为任意子集可通过结点重排变成前 m 个结点.

(ii) 注入模式为一个结点一个控制. 注入模式可能更复杂, 但在讨论控制问题时, 将本节的方法推广到更复杂的模式不会有本质困难.

记 Θ 为二元布尔算子集合或空集. 用算子的结构矩阵表示, 则

$$\Theta = \mathcal{L}_{2\times 4} \cup \varnothing.$$

这里 $\varnothing$ 表示不加控制. 设

$$\Diamond_i \in \Xi_i \subset \Theta, \quad i = 1, 2, \cdots, m.$$

$\Diamond_i = \varnothing$ 表示第 i 个牵引结点无输入控制. 记

$$\Xi_i := \left\{\xi_1^i, \xi_2^i, \cdots, \xi_{r_i}^i\right\}, \quad i = 1, 2, \cdots, m.$$

定义

$$N_j^i = \begin{cases} I_2 \otimes \mathbf{1}_2^{\mathrm{T}}, & \Diamond_i = \varnothing, \\ \Diamond_i \text{ 的结构矩阵}, & \Diamond_i \neq \varnothing, \end{cases}$$

并记

$$N^i = \left[N_1^i, N_2^i, \cdots, N_{r_i}^i\right].$$

那么, (4.3.2) 的第 i ($i \leqslant m$) 个方程的 (分量) 代数状态空间表达式为

$$\begin{aligned}
x_i(t+1) &= N^i v_i(t) M_i x(t) u_i(t) \\
&= N_i \left(\mathbf{1}_{r_1 r_2 \cdots r_{i-1}}^{\mathrm{T}} \otimes I_r \otimes \mathbf{1}_{r_{i+1} r_{i+2} \cdots r_n}^{\mathrm{T}}\right) v(t) M_i x(t) \\
&\quad \ltimes \left(\mathbf{1}_{2^{i-1}}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-i}}^{\mathrm{T}}\right) u(t) \\
&= N_i \left(\mathbf{1}_{r_1 r_2 \cdots r_{i-1}}^{\mathrm{T}} \otimes I_r \otimes \mathbf{1}_{r_{i+1} r_{i+2} \cdots r_n}^{\mathrm{T}}\right) (I_\ell \otimes M_i) \\
&\quad \ltimes \left(I_{2^n \ell} \otimes \mathbf{1}_{2^{i-1}}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-i}}^{\mathrm{T}}\right) v(t) x(t) u(t) \\
&= N_i \left(\mathbf{1}_{r_1 r_2 \cdots r_{i-1}}^{\mathrm{T}} \otimes I_r \otimes \mathbf{1}_{r_{i+1} r_{i+2} \cdots r_n}^{\mathrm{T}}\right) (I_\ell \otimes M_i) \\
&\quad \ltimes \left(I_{2^n \ell} \otimes \mathbf{1}_{2^{i-1}}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-i}}^{\mathrm{T}}\right) \left(I_\ell \otimes W_{[2^m, 2^n]}\right) v(t) u(t) x(t) \\
&:= P_i v(t) u(t) x(t).
\end{aligned} \tag{4.3.3}$$

$$P_i = N_i \left(\mathbf{1}^{\mathrm{T}}_{r_1 r_2 \cdots r_{i-1}} \otimes I_r \otimes \mathbf{1}^{\mathrm{T}}_{r_{i+1} r_{i+2} \cdots r_n}\right) (I_\ell \otimes M_i)$$
$$\ltimes \left(I_{2^n \ell} \otimes \mathbf{1}^{\mathrm{T}}_{2^{i-1}} \otimes I_2 \otimes \mathbf{1}^{\mathrm{T}}_{2^{n-i}}\right) \left(I_\ell \otimes W_{[2^m, 2^n]}\right), \quad i = 1, 2, \cdots, m.$$

对后 $n-m$ 个方程, 有

$$\begin{aligned} x_i(t+1) &= M_i x(t) = M_i \left(\mathbf{1}^{\mathrm{T}}_{2^m \ell} \otimes 2^n\right) v(t) u(t) x(t) \\ &= P_i v(t) u(t) x(t), \end{aligned} \tag{4.3.4}$$

这里

$$P_i = M_i \left(\mathbf{1}^{\mathrm{T}}_{2^m \ell} \otimes I_{2^n}\right), \quad i = m+1, m+2, \cdots, n.$$

综上可知, (4.3.2) 的第 i 个方程的 (分量) 代数状态空间表达式, 当 $i \leqslant m$ 时为 (4.3.3), 当 $i > m$ 时为 (4.3.4). 因此, (4.3.2) 的 (全局) 代数状态空间表达式为

$$x(t+1) = P v(t) u(t) x(t), \tag{4.3.5}$$
$$P = P_1 * P_2 * \cdots * P_n.$$

下面给出一个例子.

例 4.3.1　考察如下牵制控制布尔网络 (图 4.3.1):

$$\begin{cases} X_1(t+1) = (X_1(t) \vee X_3(t)) \Diamond_1(t) U_1(t), \\ X_2(t+1) = (X_2(t) \wedge X_3(t)) \Diamond_2(t) U_2(t), \\ X_3(t+1) = X_1(t) \leftrightarrow X_3(t), \end{cases} \tag{4.3.6}$$

这里 $\Diamond_1 \in \Xi_1 = \{\bar{\vee}, \rightarrow\}$, $\Diamond_2 \in \Xi_2 = \{\varnothing, \rightarrow, \vee\}$.

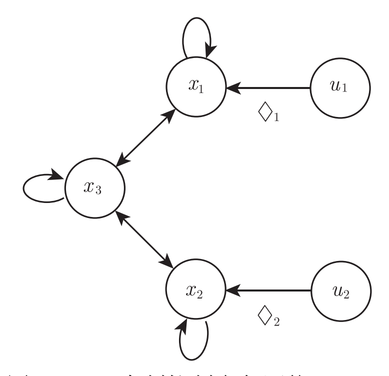


图 4.3.1　牵制控制布尔网络 (4.3.6)

分别记 $\bar{\vee}$, $\rightarrow$, $\vee$, $\wedge$ 的结构矩阵为 M_p, M_i, M_d, M_c.

(i) 如果 $\Diamond_1(t)=\bar{\vee}$, 那么, $v_1=\delta_2^1$, 于是有

$$X_1(t+1)=(X_1(t)\vee X_3(t))\bar{\vee}U_1(t). \tag{4.3.7}$$

其代数状态空间表达式为

$$\begin{aligned}x_1(t+1)&=M_pu_1(t)M_dx_1(t)x_3(t)\\&=M_p(I_2\otimes\mathbf{1}_2^{\mathrm{T}})u(t)M_d(I_2\otimes\mathbf{1}_2^{\mathrm{T}}\otimes I_2)x(t)\\&=M_p(I_2\otimes\mathbf{1}_2^{\mathrm{T}})\left\{I_4\otimes\left[M_d(I_8\otimes\mathbf{1}_2^{\mathrm{T}}\otimes I_2)\right]\right\}u(t)x(t)\\&:=M_1^1u(t)x(t),\end{aligned} \tag{4.3.8}$$

这里 $x_i=\vec{X}_i$, $u_i=\vec{U}_i$, $x(t)=\ltimes_{i=1}^3x_i(t)$, $u(t)=u_1(t)u_2(t)$, 并且

$$\begin{aligned}M_1^1&=M_p(I_2\otimes\mathbf{1}_2^{\mathrm{T}})\left\{I_4\otimes\left[M_d(I_8\otimes\mathbf{1}_2^{\mathrm{T}}\otimes I_2)\right]\right\}\\&=\delta_2[2,2,2,2,2,1,2,1,2,2,2,2,2,1,2,1,\\&\qquad 1,1,1,1,1,2,1,2,1,1,1,1,1,2,1,2].\end{aligned}$$

(ii) 如果 $\Diamond_1(t)=\to$, 即, $v_1=\delta_2^2$, 类似地可得到

$$x_1(t+1)=M_1^2u(t)x(t), \tag{4.3.9}$$

这里

$$\begin{aligned}M_1^2&=M_i(I_2\otimes\mathbf{1}_2^{\mathrm{T}})\left\{I_4\otimes\left[M_d(I_8\otimes\mathbf{1}_2^{\mathrm{T}}\otimes I_2)\right]\right\}\\&=\delta_2[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,\\&\qquad 2,2,2,2,2,1,2,1,2,2,2,2,2,1,2,1].\end{aligned}$$

令

$$M_1=[M_1^1,M_1^2],$$

于是可得

$$x_1(t+1)=M_1v_1(t)u(t)x(t). \tag{4.3.10}$$

下面考虑 $x_2(t+1)$. 类似的讨论可知:

(i) 如果 $\Diamond_2(t)=\varnothing$, 即, $v_2=\delta_3^1$, 则有

$$X_2(t+1)=X_2(t)\wedge X_3(t). \tag{4.3.11}$$

其代数形式为

$$x_2(t+1) = M_c x_2(t) x_3(t) = M_2^1 u(t) x(t), \tag{4.3.12}$$

这里

$$\begin{aligned} M_2^1 &= M_c\left(\mathbf{1}_{2^3}^{\mathrm{T}} \otimes I_4\right), \\ &= \delta_2[1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2, \\ &\qquad 1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2]. \end{aligned}$$

(ii) 如果 $\Diamond_2(t) = \rightarrow$, 即, $v_2 = \delta_3^2$, 则有

$$X_2(t+1) = (X_2(t) \wedge X_3(t)) \rightarrow U_2(t). \tag{4.3.13}$$

其代数形式为

$$\begin{aligned} x_2(t+1) &= M_i M_c x_2(t) x_3(t) u_2(t) \\ &= M_i M_c W_{[2,4]} u_2(t) x_2(t) x_3(t) \\ &= M_2^2 u(t) x(t), \end{aligned} \tag{4.3.14}$$

这里

$$\begin{aligned} M_2^2 &= M_i M_c\left(\mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}} \otimes I_4\right) \\ &= \delta_2[1,1,1,1,1,1,1,1,2,1,1,1,2,1,1,1, \\ &\qquad 1,1,1,1,1,1,1,1,2,1,1,1,2,1,1,1]. \end{aligned}$$

(iii) 如果 $\Diamond_2(t) = \vee$, 即, $v_2 = \delta_3^3$, 则有

$$X_2(t+1) = (X_2(t) \wedge X_3(t)) \vee U_2(t). \tag{4.3.15}$$

其代数形式为

$$\begin{aligned} x_2(t+1) &= M_d u_2(t) M_c x_2(t) x_3(t) \\ &= M_d(I_2 \otimes M_c) u_2(t) x_2(t) x_3(t) \\ &= M_2^3 u(t) x(t), \end{aligned} \tag{4.3.16}$$

这里

$$\begin{aligned} M_2^3 &= M_d(I_2 \otimes M_c)\left(\mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}} \otimes I_4\right) \\ &= \delta_2[1,1,1,1,1,1,1,1,1,2,2,2,1,2,2,2, \\ &\qquad 1,1,1,1,1,1,1,1,1,2,2,2,1,2,2,2]. \end{aligned}$$

令

$$M_2 = [M_2^1, M_2^2, M_2^3],$$

则

$$x_2(t+1) = M_2 v_2(t) u(t) x(t). \tag{4.3.17}$$

关于 X_3 的代数形式, 我们有

$$x_3(t+1) = M_e x_1(t) x_3(t). \tag{4.3.18}$$

将 (4.3.10), (4.3.17), 以及 (4.3.18) 写成标准形式.

(i) (4.3.10) 可表示为

$$x_1(t+1) = P_1 v(t) u(t) x(t), \tag{4.3.19}$$

这里, $v(t) = v_1(t) v_2(t)$, $u(t) = u_1(t) u_2(t)$, $x(t) = \ltimes_{i=1}^{3} x_i(t)$, 且

$$P_1 = M_1 \left(I_2 \otimes \mathbf{1}_3^{\mathrm{T}} \right). \tag{4.3.20}$$

(ii) (4.2.26) 可表示为

$$x_2(t+1) = P_2 v(t) u(t) x(t), \tag{4.3.21}$$

这里

$$P_2 = M_2 \left(\mathbf{1}_2^{\mathrm{T}} \otimes I_3 \right). \tag{4.3.22}$$

(iii) (4.3.18) 可表示为

$$x_3(t+1) = P_3 v(t) u(t) x(t), \tag{4.3.23}$$

这里

$$P_3 = M_e \left(\mathbf{1}_{24}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}} \otimes I_2 \right). \tag{4.3.24}$$

(4.3.19), (4.3.21), 以及 (4.3.23) 分别为 $X_1(t)$, $X_2(t)$, $X_3(t)$ 的分量代数状态空间表示. 利用它们即可得到网络 (4.3.6) 的代数状态空间表示:

$$x(t+1) = P v(t) u(t) x(t), \tag{4.3.25}$$

这里

$$\begin{aligned} P &= P_1 * P_2 * P_3 \\ &= \delta_8[5,8,7,8,6,3,8,3,5,8,7,8,6,3,8,3,1,4,3,4,2,7,4,7,1,4,3,4,2,7,4,7, \end{aligned}$$

$$5,6,5,6,6,1,6,1,7,6,5,6,8,1,6,1,1,2,1,2,2,5,2,5,3,2,1,2,4,5,2,5,$$
$$5,6,5,6,6,1,6,1,5,8,7,8,6,3,8,3,1,2,1,2,2,5,2,5,1,4,3,4,2,7,4,7,$$
$$1,4,3,4,2,3,4,3,1,4,3,4,2,3,4,3,5,8,7,8,6,3,8,3,5,8,7,8,6,3,8,3,$$
$$1,2,1,2,2,1,2,1,3,2,1,2,4,1,2,1,5,6,5,6,6,1,6,1,7,6,5,6,8,1,6,1,$$
$$1,2,1,2,2,1,2,1,1,4,3,4,2,3,4,3,5,6,5,6,6,1,6,1,5,8,7,8,6,3,8,3]. \tag{4.3.26}$$

4.3.2　牵制控制网络的能控性

与普通布尔控制网络不同, 牵制控制网络的能控性依赖于牵制控制的注入模式. 下面我们依注入模式考虑其能控性.

4.3.2.1　时变注入模式

如果是时变注入, 那么, 只要把注入模式看作独立的控制, 则系统 (4.3.2) 可看成具有控制 $\{\Diamond_i, U_j, i,j=1,2,\cdots,m\}$ 的混合值逻辑网络即可.

考察牵制控制网络 (4.3.2), 其代数状态空间表达式为 (4.3.5). 定义其转移矩阵为

$$T=\sum_{i=1}^{\ell}{}_{\mathcal{B}}\sum_{j=1}^{2^m}{}_{\mathcal{B}}P\delta_{\ell}^{i}\delta_{2^m}^{j}. \tag{4.3.27}$$

利用 T, 定义能控性矩阵为

$$\mathcal{C}:=\sum_{i=1}^{2^n}{}_{\mathcal{B}}T^{(i)}. \tag{4.3.28}$$

根据定理 4.1.1 的证明, 不难验证以下结论.

推论 4.3.1　考察牵制控制布尔网络 (4.3.2), 其能控性矩阵由 (4.3.28) 确定. 设注入模式是时变的, 则定理 4.1.1 的结论同样有效.

例 4.3.2　回忆例 4.3.1. 根据公式 (4.3.27) 并利用 (4.3.26), 容易算出

$$T=\begin{bmatrix}1&0&1&0&0&1&0&1\\0&1&0&1&1&0&1&0\\0&0&1&0&0&1&0&1\\0&1&0&1&0&0&1&0\\1&0&1&0&0&0&0&0\\0&1&0&1&1&0&1&0\\0&0&1&0&0&0&0&0\\0&1&0&1&0&0&1&0\end{bmatrix}. \tag{4.3.29}$$

再利用 (4.3.28) 可知

$$\mathcal{C} = \mathbf{1}_{8\times 8}. \tag{4.3.30}$$

因此, 牵制系统 (4.3.2) 在时变注入模式下是完全能控的.

4.3.2.2 定常注入模式

时变注入模式虽然在理论分析上比较简单, 但是在工程实现上则难度较大, 故需考虑定常注入模式. 在定常注入模式下注入模式虽可设计, 但是一旦选定, 则在网络演化过程中不再变化.

考察牵制控制布尔网络 (4.3.2), 其代数状态空间表达式为 (4.3.5). 考虑其在定常注入模式下的能控性. 这时, 注入模式可看作一种网络输入, 其输入网络方程为

$$v(t+1) = v(t). \tag{4.3.31}$$

将它写成标准形式为

$$\begin{aligned} v(t+1) &= \left(\mathbf{1}_{2^m}^{\mathrm{T}} \otimes I_\ell \otimes \mathbf{1}_{2^n}^{\mathrm{T}}\right) u(t)v(t)x(t) \\ &:= M_1 u(t)v(t)x(t), \end{aligned} \tag{4.3.32}$$

这里

$$M_1 = \left(\mathbf{1}_{2^m}^{\mathrm{T}} \otimes I_\ell \otimes \mathbf{1}_{2^n}^{\mathrm{T}}\right).$$

再考虑 (4.1.6), 将它写成混合输入的标准形式 (见 (4.2.10))

$$\begin{aligned} x(t+1) &= Pv(t)u(t)x(t) \\ &= PW_{[2^m,\ell]}u(t)v(t)x(t) \\ &:= M_2 u(t)v(t)x(t), \end{aligned} \tag{4.3.33}$$

这里

$$M_2 = PW_{[2^m,\ell]}.$$

将 (4.3.32) 和 (4.3.33) 放到一起, 则得到辅助系统

$$w(t+1) = Mu(t)w(t), \tag{4.3.34}$$

这里 $w(t) = v(t)x(t)$,

$$M = M_1 * M_2.$$

依 (4.2.14) 构造等价类, 则可得到

$$J_0 = J_d := J = \mathbf{1}_\ell \otimes I_{2^n}. \tag{4.3.35}$$

设辅助系统 (4.3.34) 的能控性矩阵为 $\mathcal{C}$, 则 (4.3.34) 关于 (4.3.35) 的集合能控性矩阵为

$$\mathcal{C}_S = J^{\mathrm{T}} \mathcal{C} J. \tag{4.3.36}$$

根据定理 4.2.2, (4.3.34) 关于 (4.3.35) 的集合能控性完全由集合能控性矩阵 (4.3.36) 决定.

根据构造, 下面的结论是显见的.

定理 4.3.1　考察牵制控制布尔网络 (4.3.2), 它在定常注入模式下的能控性完全由辅助系统 (4.3.34) 关于 (4.3.35) 的集合能控性矩阵 $\mathcal{C}_S = (c_{i,j})$ 确定, 即

(i) 牵制控制布尔网络 (4.3.2) 在定常注入模式下可从 X^0 控制到 X^d, 当且仅当,

$$c_{i,j} = 1.$$

(ii) 牵制控制布尔网络 (4.3.2) 在定常注入模式下在 X^0 可控, 当且仅当,

$$\mathrm{Col}_j(\mathcal{C}_S) = \mathbf{1}_{2^n}.$$

(iii) 牵制控制布尔网络 (4.3.2) 在定常注入模式下到 X^d 可达, 当且仅当,

$$\mathrm{Row}_i(\mathcal{C}_S) = \mathbf{1}^{\mathrm{T}}_{2^n}.$$

(iv) 牵制控制布尔网络 (4.3.2) 在定常注入模式下完全可控, 当且仅当,

$$\mathcal{C}_S = \mathbf{1}_{2^n \times 2^n}.$$

例 4.3.3　回忆例 4.3.1. 设注入模式为定常注入. 我们构造辅助系统:

$$\begin{aligned} v(t+1) &= M_1 u(t) v(t) x(t), \\ x(t+1) &= M_2 u(t) v(t) x(t), \end{aligned} \tag{4.3.37}$$

这里

$$M_1 = \mathbf{1}^{\mathrm{T}}_{2^m} \otimes I_6 \otimes \mathbf{1}^{\mathrm{T}}_8.$$

利用 (4.3.26),

$$
\begin{aligned}
M_2 &= PW_{[4,6]}\\
&= \delta_8[5,8,7,8,6,3,8,3,5,6,5,6,6,1,6,1,5,6,5,6,6,1,6,1,1,4,3,4,2,3,4,3,\\
&\qquad 1,2,1,2,2,1,2,1,1,2,1,2,2,1,2,1,5,8,7,8,6,3,8,3,7,6,5,6,8,1,6,1,\\
&\qquad 5,8,7,8,6,3,8,3,1,4,3,4,2,3,4,3,3,2,1,2,4,1,2,1,1,4,3,4,2,3,4,3,\\
&\qquad 1,4,3,4,2,7,4,7,1,2,1,2,2,5,2,5,1,2,1,2,2,5,2,5,5,8,7,8,6,3,8,3,\\
&\qquad 5,6,5,6,6,1,6,1,5,6,5,6,6,1,6,1,1,4,3,4,2,7,4,7,3,2,1,2,4,5,2,5,\\
&\qquad 1,4,3,4,2,7,4,7,5,8,7,8,6,3,8,3,7,6,5,6,8,1,6,1,5,8,7,8,6,3,8,3].
\end{aligned}
$$

利用 M_1, M_2, 可依次计算

$$
\begin{gathered}
M = M_1 * M_2,\\
\mathcal{C} = \sum_{i=1}^{6}{}_{\mathcal{B}} \sum_{j=1}^{4}{}_{\mathcal{B}} M\delta_6^i\delta_4^j,\\
J = \mathbf{1}_6 \otimes I_8.
\end{gathered}
$$

最后可得

$$
\mathcal{C}_S = J^{\mathrm{T}}\mathcal{C}J = \mathbf{1}_{8\times 8}.
$$

根据定理 4.3.1 可知, 牵制控制布尔网络 (4.3.6) 在定常注入模式下完全可控.

第 5 章　逻辑动态系统的能观性

能控性和能观测性 (简称能观性) 是一个控制系统的两个最基本的问题. 一个 n 维控制系统通常有 m $(< n)$ 个输出方程 (亦称观测方程), 例如系统 (4.1.2), 能观性指由输出 (或曰观测) 的数据能够确定状态. 与能控性不同的是, 逻辑动态系统的能观性有多种不等价的定义, 它们反映了对状态不同的确定程度.

5.1　四种不同的能观性

讨论布尔网络能观性的文献很多, [144] 对文献中较常见的四种能观性作了较系统的分析和比较, 亦可见 [145]. 本节内容主要来自 [144].

考察布尔控制网络 (4.1.2), 其代数状态空间表达式为 (4.1.7).

下面给出四种不同的能观性定义.

O_{I}.

定义 5.1.1 [144]　布尔控制网络 (4.1.2) 称为能观的, 如果对任意两个初始值 $X_0 \neq \bar{X}_0$ 存在一个控制序列 $\{U(0), U(1), \cdots\}$, 使相应输出序列不相同, 即

$$(Y(0), Y(1), \cdots) \neq (\bar{Y}(0), \bar{Y}(1), \cdots). \tag{5.1.1}$$

O_{II}.

定义 5.1.2 [18]　布尔控制网络 (4.1.2) 称为能观的, 如果对任一初始值 X_0 存在一个控制序列 $\{U(0), U(1), \cdots\}$, 使相应输出序列 $\{Y(0), Y(1), \cdots\}$ 能唯一确定 X_0.

注 5.1.1 [144]　定义 5.1.2 等价于以下定义: 布尔控制网络 (4.1.2) 称为能观的, 如果对任一初始值 X_0 存在一个控制序列 $\{U(0), U(1), \cdots\}$, 使对任意 $\bar{X}_0 \neq X_0$, 关于 X_0 与 $\bar{X}_0$ 相应输出序列不等, 即不等式 (5.1.1) 成立.

O_{III}.

定义 5.1.3 [80]　布尔控制网络 (4.1.2) 称为能观的, 如果存在一个控制序列 $\{U(0), U(1), \cdots\}$, 使对任意两个初始值 $X_0 \neq \bar{X}_0$ 相应输出序列不相同, 即不等式 (5.1.1) 成立.

O_{IV}.

定义 5.1.4 [44]　布尔控制网络 (4.1.2) 称为能观的, 如果对任意两个初始值 $X_0 \neq \bar{X}_0$ 及任意一个控制序列 $\{U(0), U(1), \cdots\}$, 不同初始值相应输出序列不相

同, 即不等式 (5.1.1) 成立.

接下来讨论这四种能观性的关系. 下面这个结论直接来自定义.

定理 5.1.1

$$O_{\rm IV} \Rightarrow O_{\rm III} \Rightarrow O_{\rm II} \Rightarrow O_{\rm I}. \tag{5.1.2}$$

注 5.1.2 根据文献 [144], 反向蕴涵均不成立, 即

$$O_{\rm IV} \not\Leftarrow O_{\rm III} \not\Leftarrow O_{\rm II} \not\Leftarrow O_{\rm I}. \tag{5.1.3}$$

要证明反向蕴涵均不成立, 只需分别对每一种情况举出一个反例就行了. 例如, 要证明 $O_{\rm III} \not\Leftarrow O_{\rm IV}$, 考虑如下系统

$$\begin{cases} x_1(t+1) = M_1 u(t) x(t), \\ x_2(t+1) = M_2 u(t) x(t), \end{cases} \\ y(t) = Hx(t), \tag{5.1.4}$$

这里

$$\begin{aligned} M_1 &= \delta_2[1,2,1,2,1,2,1,1], \\ M_2 &= \delta_2[1,1,1,1,1,1,2,2], \\ H &= \delta_2[1,1,2,2]. \end{aligned}$$

如果取控制 $u(0) = \delta_2^1$, 那么, 四个不同初值的输出为

$$\begin{aligned} x(0) = \delta_4^1: &\quad y(0) = \delta_2^1, y(1) = \delta_2^1, \cdots; \\ x(0) = \delta_4^2: &\quad y(0) = \delta_2^1, y(1) = \delta_2^2, \cdots; \\ x(0) = \delta_4^3: &\quad y(0) = \delta_2^2, y(1) = \delta_2^1, \cdots; \\ x(0) = \delta_4^4: &\quad y(0) = \delta_2^2, y(1) = \delta_2^2, \cdots. \end{aligned}$$

因此, (5.1.4) 在 $O_{\rm III}$ 意义下能观.

如果取控制 $u(t) = \delta_2^2$, $t = 0, 1, \cdots$, 那么, 不难看出

$$\begin{aligned} x(0) = \delta_4^3: &\quad y(0) = \delta_2^1, y(1) = \delta_2^2, y(3) = \delta_2^1, y(4) = \delta_2^2, \cdots; \\ x(0) = \delta_4^4: &\quad y(0) = \delta_2^1, y(1) = \delta_2^2, y(3) = \delta_2^1, y(4) = \delta_2^2, \cdots. \end{aligned}$$

即对此二初始值及这个控制序列, 输出均为 $y(2k) = \delta_2^1$, $y(2k+1) = \delta_2^2$, $k = 0, 1, \cdots$. 因此, 这两个初始值在这个控制序列下是无法区别的. 也就是说, (5.1.4) 在 $O_{\rm IV}$ 意义下不能观.

但构造证明另外两个反向蕴涵不成立的反例不是很容易, 这里略去.

图 5.1.1 显示了它们之间的关系 (5.1.2)—(5.1.3).

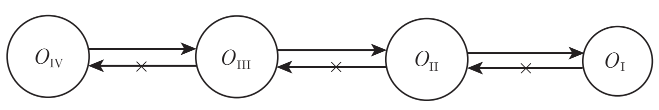


图 5.1.1 四种能观性的关系

5.2　能观性的判定

由定理 5.1.1 可知, $O_{\rm I}$ 是最敏感的能观性. 所谓最敏感是指它所要求的条件最弱. 因此, 本章的余下部分只讨论这种能观性. 关于其他各种能观性问题可参考文献 [144].

5.2.1　输出序列的直接验证

本节的内容主要来自文献 [18], 亦可参考 [22].

考察布尔控制网络 (4.1.1), 其代数状态空间表达式为 (4.1.7).

先给出一个算法, 它本质上是输出序列的列表法.

算法 5.2.1　第一步: 构造一族逻辑矩阵 $\Gamma_i \subset \mathcal{L}_{2^n\times 2^n}$, $i=1,2,\cdots$ 如下:

$$
\begin{aligned}
\Gamma_0 &= \{I_{2^n}\};\\
\Gamma_1 &= \left\{M\delta_{2^m}^i \,|\, i=1,2,\cdots,2^m\right\};\\
\Gamma_{k+1} &= \left\{M\delta_{2^m}^i\xi^k \,|\, \xi^k\in\Gamma_k;\ i=1,2,\cdots,2^m\right\},\quad k\geqslant 1.
\end{aligned}
\tag{5.2.1}
$$

注意, Γ_k 中元素个数为

$$|\Gamma_k| = 2^{km},$$

将 Γ_k, $k=0,1,\cdots$ 元素依序排成一个矩阵如 (5.2.2) 所示.

$$
\Gamma := \begin{bmatrix}
\left.I_{2^n}\right\}\Gamma_0\\
\left.\begin{matrix} M\delta_{2^m}^1\\ M\delta_{2^m}^2\\ \vdots\\ M\delta_{2^m}^{2^m}\end{matrix}\right\}\Gamma_1\\
\left.\begin{matrix} M\delta_{2^m}^1M\delta_{2^m}^1\\ \vdots\\ M\delta_{2^m}^{2^m}M\delta_{2^m}^1\\ \vdots\\ M\delta_{2^m}^1M\delta_{2^m}^{2^m}\\ \vdots\\ M\delta_{2^m}^{2^m}M\delta_{2^m}^{2^m}\end{matrix}\right\}\Gamma_2\\
\vdots
\end{bmatrix}
:= \begin{bmatrix}
\Gamma_0\\ \Gamma_1^1\\ \vdots\\ \Gamma_1^{2^m}\\ \Gamma_2^1\\ \vdots\\ \Gamma_2^{2^{2m}}\\ \vdots
\end{bmatrix}.
\tag{5.2.2}
$$

第二步: 将矩阵 Γ 每一个 $2^n \times 2^n$ 的块左乘 H, 于是

$$H\Gamma_s^r \in \mathcal{L}_{2^p \times 2^n}, \quad s = 0, 1, 2, \cdots, \quad r = 1, 2, \cdots, 2^{sn}.$$

将其写成逻辑矩阵紧致形式

$$H\Gamma_s^r = \delta_{2^p}\left[h_1^{s,r}, h_2^{s,r}, \cdots, h_{2^n}^{s,r}\right], \quad r = 1, 2, \cdots, 2^{sm}.$$

第三步: 构造能观性矩阵

$$\mathcal{O} := \begin{bmatrix} h^{0,1} \\ h^{1,1} \\ \vdots \\ h^{1,2^m} \\ h^{2,1} \\ \vdots \\ h^{2,2^{2m}} \\ \vdots \\ h^{k^*,1} \\ \vdots \\ h^{k^*,2^{k^*m}} \end{bmatrix} = \begin{bmatrix} h_1^{0,1} & h_2^{0,1} & \cdots & h_{2^n}^{0,1} \\ h_1^{1,1} & h_2^{1,1} & \cdots & h_{2^n}^{1,1} \\ \vdots & \vdots & & \vdots \\ h_1^{1,2^m} & h_2^{1,2^m} & \cdots & h_{2^n}^{1,2^m} \\ h_1^{2,1} & h_2^{2,1} & \cdots & h_{2^n}^{2,1} \\ \vdots & \vdots & & \vdots \\ h_1^{2,2^{2m}} & h_2^{1,2^{2m}} & \cdots & h_{2^n}^{2,2^{2m}} \\ \vdots & \vdots & & \vdots \\ h_1^{k^*,1} & h_2^{k^*,1} & \cdots & h_{2^n}^{k^*,1} \\ \vdots & \vdots & & \vdots \\ h_1^{k^*,2^{k^*m}} & h_2^{k^*,2^{k^*m}} & \cdots & h_{2^n}^{k^*,2^{k^*m}} \end{bmatrix}. \tag{5.2.3}$$

注 5.2.1 一个要点是, 这个算法何时停止? 即 $k^* =$? 实际上, 算法在两种情况下停下来:

(i) 能观性矩阵 $\mathcal{O}$ 对某个 k^* 任何两列都不相同. 这时 k^* 可能很小.

(ii) k^* 之后, 没有新的行出现. 因为所有可能的行是一个有限集, 因此, 有限的 (最小)k^* 一定存在. 一个保守的估计是 $k^* \leqslant 2^{2n}$. 这是因为, 从任意两点出发, 经 2^n 步, 如果不能区分, 证明一步可达集无法由输出区分. 再经 2^n 步, 如果不能区分, 证明两步可达集无法由输出区分, 如此等等, 到 $(2^n)^2 = 2^{2n}$, 如果还不能区分, 证明 2^n 步可达集无法由输出区分. 而这时, 根据能控性定理, 所有可达集均已走遍, 而相应的输出仍无法区别此二点. 于是, 此二点永远无法区别.

但这要求检验的行数为

$$1 + 2^m + 2^{2m} + \cdots + 2^{m\times 2^n} = \frac{(2^m)^{2^n+2} - 1}{2^m - 1}.$$

此数过大, 实际上很难应用. 因此, 可以把情况 (i) 看作一个合理的能观性的充分条件.

由能观性矩阵 $\mathcal{O}$ 的构造不难得到如下结论.

定理 5.2.1　考察布尔控制网络 (4.1.2), 其代数状态空间表达式为 (4.1.7). 网络 (4.1.2) 是完全可观的, 当且仅当, 能观性矩阵 $\mathcal{O}$ 的任意两列均不相同.

证明　从构造不难看出, $\mathrm{Col}_i(\mathcal{O})$ 就是 $x_0 = \delta_{2^n}^i$ 在所有可能控制下的输出, 结论显见. □

下面给出一个例子.

例 5.2.1　考察布尔控制网络

$$\begin{cases} X_1(t+1) = X_2(t) \leftrightarrow X_3(t), \\ X_2(t+1) = X_3(t) \vee U_1(t), \\ X_3(t+1) = X_1(t) \wedge U_2(t), \end{cases}$$
$$\begin{aligned} &Y_1(t) = X_1(t), \\ &Y_2(t) = X_2(t) \vee X_3(t). \end{aligned} \tag{5.2.4}$$

记其代数状态空间表示为

$$\begin{aligned} x(t+1) &= Mu(t)x(t), \\ y(t) &= Hx(t). \end{aligned} \tag{5.2.5}$$

容易算得

$$\begin{aligned} M = \delta_8[&1,5,5,1,2,6,6,2,2,6,6,2,2,6,6,2, \\ &1,7,5,3,2,8,6,4,2,8,6,4,2,8,6,4], \\ H = \delta_4[&1,1,1,2,3,3,3,4]. \end{aligned}$$

于是, 不难得到 Γ_1 如下:

$$\Gamma_1 = \begin{bmatrix} M\delta_4^1 \\ M\delta_4^2 \\ M\delta_4^3 \\ M\delta_4^4 \end{bmatrix} = \begin{bmatrix} 1,5,5,1,2,6,6,2 \\ 2,6,6,2,2,6,6,2 \\ 1,7,5,3,2,8,6,4 \\ 2,8,6,4,2,8,6,4 \end{bmatrix}.$$

进而有

$$\begin{aligned} HM\delta_4^1 &= \delta_4[1,3,3,1,1,3,3,1]; \\ HM\delta_4^2 &= \delta_4[1,3,3,1,1,3,3,1]; \\ HM\delta_4^3 &= \delta_4[1,3,3,1,1,4,3,2]; \\ HM\delta_4^4 &= \delta_4[1,4,3,2,1,4,3,2]. \end{aligned}$$

最后可得

$$\mathcal{O}=\begin{bmatrix} H\\ HM\delta_4^1\\ HM\delta_4^2\\ HM\delta_4^3\\ HL\delta_4^4\\ \vdots \end{bmatrix}=\begin{bmatrix} 1&1&1&2&3&3&3&4\\ 1&3&3&1&1&3&3&1\\ 1&3&3&1&1&4&3&2\\ 1&4&3&2&1&4&3&2\\ \vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots \end{bmatrix}.$$

由此可见 $\mathcal{O}$ 各列两两不相同, 因此, 布尔控制网络 (5.2.4) 是能观的.

5.2.2 演化点对分析

本节提出的方法是基于演化点对的分析, 内容主要来自文献 [36].

回忆布尔控制网络 (4.1.1), 因为能观性是指每个初始点对 $X_0 \neq \bar{X}_0$ 能否被区分开, 所以, 直接讨论点对是一个合理的想法. 记 $\mathcal{X}=\mathcal{D}_2^n$ 为 (4.1.1) 的状态空间. 将点对 $(X,\bar{X})\in\mathcal{X}\times\mathcal{X}$ 进行分类.

定义 5.2.1 考察布尔控制网络 (4.1.1), 其代数状态空间表达式为 (4.1.7).

(i) $(X,\bar{X})\in\mathcal{X}\times\mathcal{X}$ 称为对角对, 如果 $X=\bar{X}$. 对角对集合记作 D, 即

$$D=\left\{(X,\bar{X})\in\mathcal{X}\times\mathcal{X}\mid X=\bar{X}\right\}.$$

(ii) $(X,\bar{X})\in\mathcal{X}\times\mathcal{X}$ 称为一次输出不可区分对, 如果 $X\neq\bar{X}$, 但 $Hx=H\bar{x}$, 这里 $x_i=\vec{X}_i$, $x=\ltimes_{i=1}^n x_i$, $\bar{x}_i=\vec{\bar{X}}_i$, $\bar{x}=\ltimes_{i=1}^n\bar{x}_i$. 一次输出不可区分对集合记作 Ξ, 即

$$\Xi=\left\{(X,\bar{X})\in\mathcal{X}\times\mathcal{X}\mid X\neq\bar{X},\ \text{且}\ Hx=H\bar{x}\right\}.$$

(iii) $(X,\bar{X})\in\mathcal{X}\times\mathcal{X}$ 称为一次输出可区分对, 如果 $Hx\neq H\bar{x}$. 一次输出可区分对集合记作 Θ, 即

$$\Theta=\left\{(X,\bar{X})\in\mathcal{X}\times\mathcal{X}\mid\ Hx\neq H\bar{x}\right\}.$$

显然, 对角对 D, 一次输出不可区分对 Ξ, 以及一次输出可区分对 Θ, 构成点对集合的一个分割, 即

$$\mathcal{X}\times\mathcal{X}=D\uplus\Xi\uplus\Theta,$$

这里, $\uplus$ 表示不相交的并.

定义 5.2.2 一个对 $\{X,\bar{X}\}$ 称为可转移到 $\{Z,\bar{Z}\}$, 记作 $\{X,\bar{X}\}\to\{Z,\bar{Z}\}$, 如果存在一个控制 U 使得或者 $z=Lux$ 并且 $\bar{z}=Lu\bar{x}$ 或者 $z=Lu\bar{x}$ 并且 $\bar{z}=Lux$. 也就是说, 前一个点对在 U 作用下会转移到后一个点对 (这里不计点对中两个点的顺序). 用

$$w_{\{X,\bar{X}\}\to\{Z,\bar{Z}\}}$$

记可以将 $\{X, \bar{X}\}$ 转移到 $\{Z, \bar{Z}\}$ 的控制的个数, 称为转移指数.

下面考虑如何计算转移指数. 设 $x = \delta_{2^n}^p$, $\bar{x} = \delta_{2^n}^q$, $z = \delta_{2^n}^\alpha$, $\bar{z} = \delta_{2^n}^\beta$. 我们将 M 分成 2^m 相等的块如下:

$$M = [M_1, M_2, \cdots, M_{2^m}],$$

这里 M_s, $s = 1, 2, \cdots, 2^m$ 为 $2^n \times 2^n$ 矩阵. 那么, 很容易证明以下这个结果.

命题 5.2.1　$\{x, \bar{x}\}$ 可以被控制 $\delta_{2^m}^j$ 转移到 $\{z, \bar{z}\}$, 当且仅当,

$$\{\mathrm{Col}_p(M_j), \mathrm{Col}_q(M_j)\} = \left\{\delta_{2^n}^\alpha, \delta_{2^n}^\beta\right\}. \tag{5.2.6}$$

应用命题 5.2.1 很容易计算出任意给定的两个点的转移指数 $w_{\{x,\bar{x}\}\to\{z,\bar{z}\}}$.

下面考虑一个给定布尔控制网络的能观性. 设这个网络的一次输出不可区分集为

$$\Xi = \{\xi_1, \xi_2, \cdots, \xi_r\},$$

构造一个矩阵 $\mathcal{W} \in \mathcal{M}_{r\times(r+1)}$, 称为 Ξ 转移指数矩阵, 构造如下:

$$\mathcal{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,r} & w_{1,r+1} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,r} & w_{2,r+1} \\ \vdots & \vdots & & \vdots & \vdots \\ w_{r,1} & w_{r,2} & \cdots & w_{r,r} & w_{r,r+1} \end{bmatrix}, \tag{5.2.7}$$

这里,

$$w_{i,j} = w_{\xi_i\to\xi_j}, \quad j \neq r+1$$

为 ξ_i 到 ξ_j 的转移指数.

$$w_{i,r+1} = w_{\xi_i\to D}$$

是 ξ_i 到对角集的转移指数.

下面介绍指数矩阵 $\mathcal{W}$ 每一行 (即每一个点对) 的不可区分指数, 记作 d_i, 定义如下:

$$d_i = \sum_{j=1}^{r+1} w_{i,j}, \quad i = 1, \cdots, r. \tag{5.2.8}$$

实际上, 行不可区分指数也是一种转移指数, d_i 是从 ξ_i 到 $\Xi \cup D$ 的转移指数, 即

$$d_i = w_{\xi_i\to\Xi\cup D}. \tag{5.2.9}$$

利用行不可区别指数, 我们通过改造转移矩阵 $\mathcal{W}$ 构造一个新的布尔矩阵 $\mathcal{E}^0=(u_{i,j})\in\mathcal{B}_{r\times(r+1)}$. 具体构造如下:

$$e^0_{i,j}=\begin{cases}1, & w_{i,j}>0,\\ 0, & w_{i,j}=0, \quad i=1,\cdots,r, \quad j=1,\cdots,r;\end{cases}$$

并且

$$e^0_{i,r+1}=\begin{cases}1, & d_i<2^m,\\ 0, & d_i=2^m, \quad i=1,\cdots,r.\end{cases}$$

根据定义, $\mathcal{E}^0$ 显然是一个布尔矩阵.

利用 $\mathcal{E}^0$ 作为初始阵, 我们通过迭代, 构造一族矩阵 $\mathcal{E}^k$ 如下.

算法 5.2.2 设 $\mathcal{E}^k=\left(u^k_{i,j}\right)$ 已知, 我们构造 $\mathcal{E}^{k+1}$ 如下:

对每个 $i=1,\cdots,r$, 如果 $e^k_{i,r+1}=1$,

$$\mathrm{Col}_{r+1}\left(\mathcal{E}^{k+1}\right)=\mathrm{Col}_{r+1}\left(\mathcal{E}^{k}\right)+_{\mathcal{B}}\mathrm{Col}_{i}\left(\mathcal{E}^{k}\right). \tag{5.2.10}$$

如果

$$\mathcal{E}^{k^*+1}=\mathcal{E}^{k^*}, \tag{5.2.11}$$

定义

$$\mathcal{E}^*:=\mathcal{E}^{k^*}, \tag{5.2.12}$$

结束. 否则, 回到上一步, 继续构造下一个矩阵 $\mathcal{E}^{k+2}$.

注意, 每一步有效的迭代都说明等式 (5.2.11) 不成立. 于是, 最后一列作为布尔向量, 它的汉明范数 $\|\mathrm{Col}_{r+1}(\mathcal{U}^k)\|_H$ (即元素和) 至少要增加 1. 因此, 至多经 $r-1$ 次迭代, 即可达到 (5.2.12). 此后则有 $\mathcal{E}^k=\mathcal{E}^{k^*}$, $\forall\, k>k^*$. 因此, 经至多 r 次迭代即可得到 $\mathcal{E}^*$.

利用 $\mathcal{E}^*$, 就可以得到以下能观性的充要条件.

定理 5.2.2 考察布尔控制网络 (4.1.1), 其代数状态空间表达式为 (4.1.7).

(i) 网络 (4.1.2) 能观, 当且仅当,

$$\mathrm{Col}_{r+1}\left(\mathcal{E}^*\right)=\mathbf{1}_r. \tag{5.2.13}$$

(ii) $\xi_i=(x_0,\bar{x}_0)$ 是不可区分的对, 当且仅当, $\mathcal{E}^*$ 的 $(i,r+1)$ 的元素

$$e_{i,r+1}=0. \tag{5.2.14}$$

证明　先证明 (i).

(充分性) 如果 $\{x_0,\bar{x}_0\}\in\Theta$, 那么, 这对点显然是可区分的. 因此, 我们只要考虑 $\{x_0,\bar{x}_0\}\in\Xi$. 记 $\xi_i=\{x_0,\bar{x}_0\}\in\Xi$, 如果它所在行的不可区分指数 $d_i<2^m$, 因为

$$w_{\xi_i\to\Xi\cup D}+w_{\xi_i\to\Theta}=2^m,$$

那么, 至少有一个控制, 它将 ξ_i 转移进 Θ. 因此, ξ_i 是一步控制可区分的. 现在考虑矩阵 $\mathcal{H}^0$. 如果 $\xi_i,\xi_j\in\Xi$, 且 $e^0_{i,r+1}=0$, $e^0_{j,r+1}=1$, 并且 $e^0_{i,j}=1$, 那么, 由构造可知, ξ_j 是一步控制可区分的, 而 ξ_i 是一步控制不可区分的. 但 $e^0_{i,j}=1$, 这表明至少有一个控制, 它能将 ξ_i 转移到 ξ_j. 因为 ξ_j 是一步控制可区分的, 则 ξ_i 是两步控制可区分的. 因此, 在 $\mathcal{E}^1$ 中, 根据算法, 我们将 $e^0_{i,r+1}=0$ 改为 $e^1_{i,r+1}=1$. 于是可知, $e^1_{i,r+1}=1$ 表示至多两次控制下相应的 ξ_i 可区分. 依次类推可知, 只要在某一步 $\mathrm{Col}_{r+1}(\mathcal{U}^k)=\mathbf{1}_r$, 则说明所有 Ξ 中的点对, 在至多 $k+1$ 步控制后均可区分.

(必要性) 记

$$\Gamma=\left\{\xi_i\in\Xi\middle|u^*_{i,r+1}=0,\ i=1,\cdots,r\right\}.$$

那么, 显然 Γ 是一个控制不变集, 即, 如果 $\gamma\in\Gamma$, 那么

$$Mu\gamma\in\Gamma,\quad \forall u\in\varDelta_{2^m}.$$

因此, Γ 是一个不可区分对的集合. 于是 $\Gamma=\varnothing$ 是能观的必要条件, 这表明 $\mathrm{Col}_{r+1}(\mathcal{E}^*)=\mathbf{1}_r$.

从 (i) 的证明可以看出 (ii) 是显见的. □

注 5.2.2　(i) 根据定理 5.2.2, $\mathrm{Col}_{r+1}\left(\mathcal{E}^*\right)$ 的第 i 个分量表示 ξ_i 是否为可区分对. 因此, 我们把 $\mathrm{Col}_{r+1}\left(\mathcal{E}^*\right)$ 称为能观向量, 并记其为

$$V^*_d:=\mathrm{Col}_{r+1}\left(\mathcal{E}^*\right).$$

类此, 将

$$V^k_d:=\mathrm{Col}_{r+1}\left(\mathcal{E}^k\right)$$

称为 $k+1$ 步能观向量.

(ii) 在算法 5.2.2 中, 实际上我们每步只更新能观向量. 因此, 式 (5.2.11) 可以用下面的 (5.2.15) 代替:

$$\|V^{k^*+1}_d\|_H=\|V^{k^*}_d\|_H. \tag{5.2.15}$$

下面给出一个例子.

例 5.2.2 考察如下的布尔网络:

$$\begin{cases} X_1(t+1) = [U(t) \wedge \neg(X_1(t) \wedge X_2(t) \wedge X_3(t))] \vee \{\neg U(t) \wedge [(X_1(t) \\ \qquad\qquad \wedge (X_2(t) \vee \neg X_3(t))) \vee (\neg X_1(t) \wedge X_2(t) \wedge \neg X_3(t))]\}, \\ X_2(t+1) = \{U(t) \wedge [(X_1(t) \wedge X_2(t) \wedge \neg X_3(t)) \vee (\neg X_1(t) \\ \qquad\qquad \wedge X_2(t) \wedge X_3(t)) \vee \neg(X_1(t) \vee X_2(t) \vee X_3(t))]\} \vee \{\neg U(t) \\ \qquad\qquad \wedge [(X_1(t) \wedge X_2(t)) \vee (\neg X_1(t) \wedge X_2(t) \wedge X_3(t))]\}, \\ X_3(t+1) = [U(t) \wedge \neg(X_2(t) \wedge X_3(t))] \\ \qquad\qquad \vee [\neg U(t) \wedge (\neg X_1(t) \wedge \neg X_2(t) \wedge X_3(t))], \end{cases}$$
$$Y_1(t) = X_1(t) \vee \neg X_2(t) \vee X_3(t),$$
$$Y_2(t) = \neg X_1(t) \vee X_2(t) \wedge \neg X_3(t). \tag{5.2.16}$$

不难算得, 网络 (5.2.16) 的代数状态空间表示为

$$\begin{aligned} x(t+1) &= Mu(t)x(t), \\ y(t) &= Hx(t), \end{aligned} \tag{5.2.17}$$

这里

$$M = \delta_8[8,1,3,3,2,3,3,1,1,4,5,3,5,3,7,7],$$
$$H = \delta_4[2,1,2,2,2,3,2,1].$$

直接计算可得, 一次输出不可区分对为

$$\begin{aligned} \Xi = \big\{&\xi_1 = \{\delta_8^1, \delta_8^3\}, \xi_2 = \{\delta_8^1, \delta_8^4\}, \xi_3 = \{\delta_8^1, \delta_8^5\}, \xi_4 = \{\delta_8^1, \delta_8^7\}, \\ &\xi_5 = \{\delta_8^2, \delta_8^8\}, \xi_6 = \{\delta_8^3, \delta_8^4\}, \xi_7 = \{\delta_8^3, \delta_8^5\}, \xi_8 = \{\delta_8^3, \delta_8^7\}, \\ &\xi_9 = \{\delta_8^4, \delta_8^5\}, \xi_{10} = \{\delta_8^4, \delta_8^7\}, \xi_{11} = \{\delta_8^5, \delta_8^7\}\big\}. \end{aligned}$$

应用命题 5.2.1 很容易计算网络的转移矩阵 $\mathcal{W}$, 将其放入表 5.2.1 中去, 那里, 最后一列放不可区分指数 d_i.

利用 $\mathcal{W}$, 我们构造 $\mathcal{E}_0$, 见表 5.2.2. 最后一列是一步能观向量 V_d^0.

下面进行迭代计算.

第一步: 注意到 $e_{i,r+1}^0 = 1$, 这里 $i \in J_0 := \{1,2,4,7,9,11\}$, 于是有

$$\begin{aligned} V_d^1 &= V_d^0 \cup \left[\bigcup_{i \in J_0} \mathrm{Col}_i\left(\mathcal{E}^0\right)\right] \\ &= [1,1,0,1,0,1,1,1,1,0,1]^{\mathrm{T}}. \end{aligned}$$

表 5.2.1 (5.2.16) 的转移矩阵 $\mathcal{W}$

P	ξ_1	ξ_2	ξ_3	ξ_4	ξ_5	ξ_6	ξ_7	ξ_8	ξ_9	ξ_{10}	ξ_{11}	D	d_i
ξ_1	0	0	1	0	0	0	0	0	0	0	0	0	1
ξ_2	1	0	0	0	0	0	0	0	0	0	0	0	1
ξ_3	0	0	1	0	1	0	0	0	0	0	0	0	2
ξ_4	0	0	0	1	0	0	0	0	0	0	0	0	1
ξ_5	0	0	0	0	0	0	0	0	0	1	0	1	2
ξ_6	0	0	0	0	0	0	1	0	0	0	0	1	2
ξ_7	0	0	0	0	0	0	0	0	0	0	0	1	1
ξ_8	0	0	0	0	0	0	0	0	0	0	1	1	2
ξ_9	0	0	0	0	0	0	1	0	0	0	0	0	1
ξ_{10}	0	0	0	0	0	0	0	1	0	0	0	1	2
ξ_{11}	0	0	0	0	0	0	0	0	0	0	1	0	1

表 5.2.2 能观性矩阵 $\mathcal{E}_0$

P	ξ_1	ξ_2	ξ_3	ξ_4	ξ_5	ξ_6	ξ_7	ξ_8	ξ_9	ξ_{10}	ξ_{11}	V_d^0
ξ_1	0	0	1	0	0	0	0	0	0	0	0	1
ξ_2	1	0	0	0	0	0	0	0	0	0	0	1
ξ_3	0	0	1	0	1	0	0	0	0	0	0	0
ξ_4	0	0	0	1	0	0	0	0	0	0	0	1
ξ_5	0	0	0	0	0	0	0	0	0	1	0	0
ξ_6	0	0	0	0	0	0	1	0	0	0	0	0
ξ_7	0	0	0	0	0	0	0	0	0	0	0	1
ξ_8	0	0	0	0	0	0	0	0	0	0	1	0
ξ_9	0	0	0	0	0	0	1	0	0	0	0	1
ξ_{10}	0	0	0	0	0	0	0	1	0	0	0	0
ξ_{11}	0	0	0	0	0	0	0	0	0	0	1	1

第二步: 因为 $e_{i,r+1}^1 = 1$, 这里 $i \in J_1 := \{1,2,4,6,7,8,9,11\}$, 于是有

$$\begin{aligned} V_d^2 &= V_d^1 \cup \left[\bigcup_{i \in J_1} \mathrm{Col}_i \left(\mathcal{E}^1\right) \right] \\ &= [1,1,0,1,0,1,1,1,1,1,1]^{\mathrm{T}}. \end{aligned}$$

第三步: 因为 $e_{i,r+1}^2 = 1$, 这里 $i \in J_2 := \{1,2,4,6,7,8,9,10,11\}$, 于是有

$$\begin{aligned} V_d^3 &= V_d^2 \cup \left[\bigcup_{i \in J_2} \mathrm{Col}_i \left(\mathcal{E}^2\right) \right] \\ &= [1,1,0,1,1,1,1,1,1,1,1]^{\mathrm{T}}. \end{aligned}$$

第四步: 因为 $e^3_{i,r+1}=1$, 这里 $i\in J_3:=\{1,2,4,5,6,7,8,9,10,11\}$, 于是有

$$\begin{aligned} V_d^4 &= V_d^3 \cup \left[\bigcup_{i\in J_3} \mathrm{Col}_i\left(\mathcal{E}^3\right)\right] \\ &= [1,1,1,1,1,1,1,1,1,1,1]^{\mathrm{T}}. \end{aligned}$$

因为 $V_d^4=\mathbf{1}_{11}$, 可知 $V_d^*=V_d^4$. 因此, 网络 (5.2.16) 是完全能观的.

第二个例子来自生物系统, 它考虑大肠杆菌 (escherichia coli) 在乳糖操纵子 (lac operon) 作用下的演化模型.

例 5.2.3 大肠杆菌在乳糖操纵子作用下的布尔网络模型如下[114]:

$$\begin{cases} X_1(t+1)=\neg X_3(t)\wedge X_7(t)\neg X_8(t), \\ X_2(t+1)=X_1(t), \\ X_3(t+1)=\neg X_4(t)\wedge\neg X_9(t), \\ X_4(t+1)=X_2(t)\wedge X_5(t), \\ X_5(t+1)=\neg U_1(t)\wedge U_2(t)\wedge X_6(t), \\ X_7(t+1)=\neg U_1(t), \\ X_8(t+1)=X_3(t)\vee(\neg X_4(t)\wedge\neg X_9(t)), \\ X_9(t+1)=X_5(t)\vee X_{10}(t), \\ X_{10}(t+1)=\neg U_1(t)\wedge(U_2(t)\vee(U_3(t)\wedge X_6(t))). \end{cases} \tag{5.2.18}$$

$$\begin{aligned} &Y_i(t)=X_i(t), \quad i=1,2,6,7, \\ &Y_3(t)=X_3(t)\wedge X_8(t), \\ &Y_4(t)=X_4(t)\vee X_9(t), \\ &Y_5(t)=X_5(t)\wedge X_{10}(t). \end{aligned}$$

这里略去每个变量的实际物理意义 (见 [114]).

不难得到它的代数状态空间表示为

$$\begin{aligned} x(t+1)&=Mu(t)x(t), \\ y(t)&=Hx(t), \end{aligned} \tag{5.2.19}$$

这里

$$\begin{aligned} &M=\delta_{1024}[682,682,682,682,\cdots,1014,1016,882,884]\in\mathcal{L}_{1024\times 8192}; \\ &H=\delta_{128}[1,5,1,5,\cdots,120,120,128,128]\in\mathcal{L}_{128\times 1024}. \end{aligned}$$

直接计算可知, 它的输出不可区分集 Ξ 如下:

$$\begin{aligned}\Xi = \big\{&\{\delta_{1024}^1, \delta_{1024}^3\}, \{\delta_{1024}^1, \delta_{1024}^{65}\}, \{\delta_{1024}^2, \delta_{1024}^4\}, \{\delta_{1024}^2, \delta_{1024}^{33}\},\\ &\cdots, \{\delta_{1024}^{1021}, \delta_{1024}^{1022}\}, \{\delta_{1024}^{1023}, \delta_{1024}^{1024}\}\big\}.\end{aligned}$$

这里 $|\Xi| = 7488$, 不便都列出. 利用算法 5.2.2 略去迭代计算细节, 可以检验, 这个系统不是完全能观的. 记它的不可区分对集合为 Ξ_r, 则

$$\Xi_r = \big\{\{\delta_{1024}^1, \delta_{1024}^3\}, \{\delta_{1024}^1, \delta_{1024}^{65}\}, \{\delta_{1024}^2, \delta_{1024}^4\}, \{\delta_{1024}^2, \delta_{1024}^{33}\}, \cdots, \{\delta_{1024}^{1020}, \delta_{1024}^{1024}\}\big\},$$

这里, $|\Xi_r| = 2032$.

5.2.3　辅助系统方法

辅助系统方法要构造一个辅助系统, 然后, 将原系统的能观性问题转换为辅助系统的集合能控性问题. 这种方法是文献 [37] 中给出的.

回到布尔控制网络 (4.1.2), 利用其代数状态空间表达式 (4.1.7), 构造一个辅助系统如下:

$$\begin{cases} z(t+1) = Mu(t)z(t), \\ x(t+1) = Mu(t)x(t). \end{cases} \tag{5.2.20}$$

我们的目的是要将布尔控制网络 (4.1.2) 的能观性问题转换为辅助系统 (5.2.20) 的集合能控性问题. 为此, 我们构造如下的初始集与目标集:

$$P^0 := \{zx \mid (z, x) \in \Xi\}, \tag{5.2.21}$$

$$P^d := \{\Theta\}. \tag{5.2.22}$$

注意到 (5.2.21) 表示, 每一个点对 $\{z, x\} \in \Xi$ 均为 P^0 中的一个元素 $\{zx\}$, 这时, 每个 $\{zx\}$ 看成一个单点集. (5.2.22) 表示 P^d 中只有一个元素, 它就是集合 Θ.

那么, 我们有如下结论.

定理 5.2.3　布尔控制网络 (4.1.2) 是完全能观的, 当且仅当, 辅助系统 (5.2.20) 关于由 (5.2.21) 定义的初始集与 (5.2.22) 定义的目标集是集合完全能控的.

证明　(必要性) 设 (4.1.2) 完全能观. 那么, 对任何两个初始值 $z_0 \neq x_0$, 总存在一个控制序列 $\{u(t) \mid t = 0, 1, \cdots\}$, 使得对应的输出序列 $\{y(t) \mid t = 0, 1, \cdots\}$ 与 $\{\tilde{y}(t) \mid t = 0, 1, \cdots\}$ 不同. 设 $T \geqslant 0$ 是最小的 t, 使得 $y(t) \neq \tilde{y}(t)$. 如果 $T = 0$, 则 $(z_0, x_0) \in \Theta$ 是一步可区分对. 因此, 不妨设 $(z_0, x_0) \in \Xi$, 即 $Hz(0) = Hx(0)$.

应用上述控制序列于 (5.2.20), 则存在 $T > 0$, 使 z_0x_0 可以达到点 $z(T)x(T)$, 这里, $Hz(T) = y(T) \neq \tilde{y}(T) = Hx(T)$. 即, $(z(T), x(T)) \in \Theta$. 由于 (z_0, x_0) 是任选的, 可知 (5.2.20) 从 P^0 到 P^d 集合能控.

(充分性) 任给 $z_0 \neq x_0$. 如果 $(z_0, x_0) \in \Theta$, 那么, 一步输出就可将它们区分开. 因此, 只要考虑 $(z_0, x_0) \in \Xi$, 即 $\{z_0x_0\} \in P^0$. 因为辅助系统 (5.2.20) 是从 P^0 到 $P^d = \{\Theta\}$ 集合能控, 则存在控制序列 $\{u(t) \mid t = 0, 1, \cdots\}$, 它可将 z_0x_0 控制到 $z(T)x(T) \in \Theta$. 也就是说, 应用这组控制序列到 (4.1.2), 它能将初值为 z_0 与 x_0 的轨线分别控制到 $z(T)$ 与 $x(T)$, 又 $y(T) = Hz(T) \neq Hx(T) = \tilde{y}(T)$. 因此, 这两初值在该控制序列作用下的输出轨线不同. □

例 5.2.4 回忆例 5.2.3, 文献 [114] 等给出一个简化模型

$$
\begin{cases}
X_1(t+1) = \neg U_1(t) \wedge (X_2(t) \vee X_3(t)), \\
X_2(t+1) = \neg U_1(t) \wedge U_2(t) \wedge X_1(t), \\
X_3(t+1) = \neg U_1(t) \wedge (U_2(t) \vee (U_3(t) \wedge X_1(t))),
\end{cases}
\\
Y_1(t) = X_1(t) \vee \neg X_2(t) \vee X_3(t), \\
Y_2(t) = \neg X_1(t) \vee X_2(t) \wedge \neg X_3(t), \\
Y_3(t) = \neg X_1(t) \wedge \neg X_2(t) \vee X_3(t).
\tag{5.2.23}
$$

不难算出, 它的代数状态空间表达式为

$$
\begin{aligned}
x(t+1) &= Mu(t)x(t), \\
y(t) &= Hx(t),
\end{aligned}
\tag{5.2.24}
$$

这里

$$
\begin{aligned}
M = \delta_8[&8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8,8, \\
&8,8,8,8,8,8,8,8,1,1,1,5,3,3,3,7,1,1,1,5,3,3,3,7, \\
&3,3,3,7,4,4,4,8,4,4,4,8,4,4,4,8], \\
H = \delta_8[&8,6,3,6,5,6,7,6].
\end{aligned}
$$

辅助系统可构造如下:

$$
\begin{cases}
z(t+1) = Mu(t)z(t), \\
x(t+1) = Mu(t)x(t).
\end{cases}
\tag{5.2.25}
$$

对辅助系统, 不难算出

$$
\begin{aligned}
\Xi &= \big\{\{\delta_8^2, \delta_8^4\}, \{\delta_8^2, \delta_8^6\}, \{\delta_8^2, \delta_8^8\}, \{\delta_8^4, \delta_8^6\}, \{\delta_8^4, \delta_8^8\}, \{\delta_8^6, \delta_8^8\}\big\} \\
&\sim \big\{\delta_{64}^{12}, \delta_{64}^{14}, \delta_{64}^{16}, \delta_{64}^{30}, \delta_{64}^{32}, \delta_{64}^{48}\big\} \\
&:= \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6\};
\end{aligned}
$$

以及

$$\begin{aligned}\Theta = \big\{&\{\delta_8^1,\delta_8^2\},\{\delta_8^1,\delta_8^3\},\{\delta_8^1,\delta_8^4\},\{\delta_8^1,\delta_8^5\},\{\delta_8^1,\delta_8^6\},\{\delta_8^1,\delta_8^7\},\{\delta_8^1,\delta_8^8\},\\ &\{\delta_8^2,\delta_8^3\},\{\delta_8^2,\delta_8^5\},\{\delta_8^2,\delta_8^7\},\{\delta_8^4,\delta_8^3\},\{\delta_8^4,\delta_8^5\},\{\delta_8^4,\delta_8^7\},\{\delta_8^6,\delta_8^3\},\\ &\{\delta_8^6,\delta_8^5\},\{\delta_8^6,\delta_8^7\},\{\delta_8^8,\delta_8^3\},\{\delta_8^8,\delta_8^5\},\{\delta_8^8,\delta_8^7\},\{\delta_8^3,\delta_8^5\},\{\delta_8^3,\delta_8^7\},\\ &\{\delta_8^5,\delta_8^7\}\big\}\\ \sim \big\{&\delta_{64}^2,\delta_{64}^3,\delta_{64}^4,\delta_{64}^5,\delta_{64}^6,\delta_{64}^7,\delta_{64}^8,\delta_{64}^{11},\delta_{64}^{13},\delta_{64}^{15},\delta_{64}^{21},\\ &\delta_{64}^{23},\delta_{64}^{27},\delta_{64}^{29},\delta_{64}^{31},\delta_{64}^{39},\delta_{64}^{43},\delta_{64}^{45},\delta_{64}^{47},\delta_{64}^{59},\delta_{64}^{61},\delta_{64}^{63}\big\}.\end{aligned}$$

令 $w(t)=z(t)x(t)$, 那么 (5.2.25) 可表示为

$$\begin{cases} z(t+1) = M\left(I_{64}\otimes \mathbf{1}_8^{\mathrm{T}}\right)u(t)w(t),\\ x(t+1) = M\left(I_8\otimes \mathbf{1}_8^{\mathrm{T}}\right)u(t)w(t).\end{cases} \tag{5.2.26}$$

于是可得

$$w(t+1) = Lu(t)w(t), \tag{5.2.27}$$

这里

$$L = M * M = \delta_{64}[64,64,64,\cdots,60,60,60,64] \in \mathcal{L}_{64\times 512}.$$

(5.2.25) 的能控性矩阵可算得为

$$\mathcal{C} := \sum_{j=1}^{64}{}_{\mathcal{B}}\left(\sum_{i=1}^{8}{}_{\mathcal{B}} L\delta_8^i\right)^{(j)} \in \mathcal{B}_{64\times 64}.$$

根据 (5.2.21) 及 (5.2.22) 可以构造初始集与目标集. $P^0=\{\theta_1,\theta_2,\theta_3,\theta_4,\theta_5,\theta_6\}$, $P^d=\{\Xi\}$, 从而得到

$$J_0 = \delta_{64}[12,14,16,30,32,48].$$

$$J_d = \sum_{\delta_{64}^i\in\Xi}\delta_{64}^i;$$

最后可得

$$\mathcal{C}_S = J_d^{\mathrm{T}}\mathcal{C}J_0 = [1,1,1,1,1,1] > 0.$$

根据定理 5.2.3, 布尔控制网络 (5.2.24) 是完全能观的.

5.3　多值逻辑动态系统的能观性

回忆布尔控制网络的动态方程 (4.1.2).

(i) 如果 $X_i, U_\sigma, Y_j \in \mathcal{D}_k$, $k > 2$, 那么, 系统变为 k 值逻辑控制网络.

(ii) 如果 $X_i \in \mathcal{D}_{k_i}$, $i = 1, \cdots, n$, $U_\alpha \in \mathcal{D}_{r_\alpha}$, $\alpha = 1, \cdots, m$, $Y_j \in \mathcal{D}_{s_j}$, $j = 1, \cdots, p$, 只要 $\{k_i, r_\alpha, s_j \mid i = 1, \cdots, n; \alpha = 1, \cdots, m; , j = 1, \cdots, p\}$ 中有两个不相等, 那么, 系统变为混合值逻辑控制网络.

无论是 k 值逻辑控制网络还是混合值逻辑控制网络, 它们的代数状态空间表达式都形同 (4.1.7). 注意到, 在 5.2 节中介绍的三种能观性判别方法包括输出序列的直接验证法、点对动态分析方法, 以及辅助系统方法, 在应用时真正需要的是系统的代数状态空间表示. 更准确地说, 需要的是矩阵 M 和 H. 回顾相关的判别方法, 如果用 k 值或混合值逻辑网络的代数状态空间表达式中的矩阵 M 和 H 代替布尔控制网络中的矩阵 M 和 H, 所有的推理过程继续有效. 因此, 在 5.2 节中介绍的三种能观性判别方法对 k 值或混合值逻辑控制网络仍然有效, 下面用一个例子说明.

例 5.3.1　考察一个形同 (4.1.2) 的混合值逻辑系统, 这里 $X_1(t)$, $X_3(t)$, $U(t) \in \varDelta_2$, $X_2(t) \in \varDelta_3$. 设状态方程的代数状态空间表达式为

$$x(t+1) = Mu(t)x(t), \tag{5.3.1}$$

这里

$$M = \delta_{12}[2,9,5,4,7,12,9,6,7,12,3,2,9,12,9,5,8,1,12,1,10,1,6,9] \in \mathcal{L}_{12\times 24}. \tag{5.3.2}$$

(i) 设有唯一输出

$$y_1(t) = H_1 x(t), \tag{5.3.3}$$

这里

$$H_1 = \delta_3[2,1,2,3,3,2,2,1,3,2,3,1] \in \mathcal{L}_{3\times 12}. \tag{5.3.4}$$

我们用点对动态分析方法来考察它的能观性.

容易算得, 一步不可区分集为

$$\Xi = \left\{\{\delta_{12}^1, \delta_{12}^3\}, \{\delta_{12}^1, \delta_{12}^6\}, \cdots, \{\delta_{12}^8, \delta_{12}^{12}\}, \{\delta_{12}^9, \delta_{12}^{11}\}\right\},$$

这里 $|\Xi| = 19$. 一次迭代后不可区分点对数为 2. 继续迭代可得, $V_d^k = V_d^2$, $k > 2$. 并且, 二次迭代后有一个不可区分点对, 为

$$\{\delta_{12}^6, \delta_{12}^{10}\}.$$

因此, 网络 (5.3.1)—(5.3.3) 是不可观的.

(ii) 设在原系统基础上增加一个输出

$$y_2(t) = H_2 x(t), \tag{5.3.5}$$

这里

$$H_2 = \delta_2[2,1,1,1,1,1,2,1,1,2,1,1] \in \mathcal{L}_{2\times 12}. \tag{5.3.6}$$

令 $y = y_1 \ltimes y_2$, 则

$$\begin{aligned} H &= H_1 * H_2 \\ &= \delta_6[4,1,3,5,5,3,4,1,5,4,5,1] \in \mathcal{L}_{6\times 12}. \end{aligned}$$

直接计算可得

$$\Xi = \left\{\{\delta_{12}^1, \delta_{12}^7\}, \{\delta_{12}^1, \delta_{12}^{10}\}, \cdots, \{\delta_{12}^8, \delta_{12}^{12}\}, \{\delta_{12}^9, \delta_{12}^{11}\}\right\},$$

这里, $|\Xi| = 13$. 此外, 还可算得 $\mathrm{Col}_{14}(\mathcal{U}^0) = \mathbf{1}_{13}$. 因此, 增加一个观察后, 网络 (5.3.1)—(5.3.3)—(5.3.5) 是完全能观的.

第 6 章　状态空间方法与坐标变换

Kalman 在 1962 年提出了线性控制系统的状态空间方法[71], 它成为现代控制理论的三大支柱之一. 利用泛函分析与微分流形等工具, 状态空间方法被推广到分布参数系统以及非线性系统等一般控制系统中去, 从而使状态空间成为控判系统分析与综合的基本平台. 对于逻辑动态系统, 如何定义适当的状态空间, 从而引入有效的状态空间方法, 这对逻辑动态系统的分析与控制极具重要性. 本章关于逻辑系统状态空间的内容主要来自文献 [21].

6.1　布尔网络的状态空间结构

6.1.1　状态空间的矩阵表示

考察一个布尔网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t)), \end{cases} \qquad X_i \in \mathcal{D}, \tag{6.1.1}$$

或者一个布尔控制网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \end{cases} \tag{6.1.2}$$
$$Y_j = h_j(X_1(t), X_2(t), \cdots, X_n(t)), \quad j = 1, 2, \cdots, p,$$

这里, $X_i(t), U_\alpha(t), Y_j(t) \in \mathcal{D}$.

与连续状态的力学系统不同, 这里所有的变量都是逻辑变量, 所有的函数都是逻辑函数. 因此, 没有相应的向量空间当作自然的状态空间结构. 于是, 我们必须给出一个合理的定义.

定义 6.1.1 考虑布尔网络 (6.1.1) 或布尔控制网络 (6.1.2).

(i) 其状态空间, 称为布尔网络的状态空间, 记作 $\mathcal{X}$, 定义为由 $X_1, X_2, \cdots, X_n$ 的所有逻辑函数组成的集合, 记作 $\mathcal{F}_\ell\{X_1, X_2, \cdots, X_n\}$, 即

$$\mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, \cdots, X_n\}. \tag{6.1.3}$$

(ii) 设 $Z_1, Z_2, \cdots, Z_r \in \mathcal{X}$, 那么, 由 $Z_1, Z_2, \cdots, Z_r$ 生成的子空间, 称为布尔网络的状态子空间, 记作 $\mathcal{Z}$, 指 $Z_1, Z_2, \cdots, Z_r$ 的逻辑函数集合, 即

$$\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2, \cdots, Z_r\}. \tag{6.1.4}$$

利用函数集合表示空间与子空间是一种常用的方法. 例如: 在线性代数中, n 维向量空间上所有的线性函数也是一个 n 维向量空间, 称为对偶空间; 关于部分变量的线性函数构成向量子空间; 在泛函线性空间 (如 L_p) 上的线性泛函集合, 记作 L_q, 其中 $\dfrac{1}{p} + \dfrac{1}{q} = 1$, 也是一个向量空间; 等等.

用逻辑函数集合表示逻辑动态系统的状态空间及子空间, 其主要优势是: 在逻辑变量及逻辑函数的向量及代数状态空间表示下, 空间及其子空间可以与逻辑矩阵相对应. 下面讨论逻辑 (子) 空间的矩阵表示.

设 $Z \in \mathcal{X}$, 则 Z 为 $X_1, X_2, \cdots, X_n$ 的一个逻辑函数. 记

$$Z = G(X_1, X_2, \cdots, X_n).$$

于是, Z 的代数状态空间表示为

$$z = M_G \ltimes_{i=1}^n x_i,$$

这里 $M_G \in \mathcal{L}_{2\times 2^n}$ 是函数 G 的结构矩阵. 注意到 M_G 可表示为

$$\delta_2[i_1, i_2, \cdots, i_{2^n}],$$

这里 i_s 可以等于 1 或 2. 因此, 可以有 2^{2^n} 个不同的逻辑函数, 从而有

$$|\mathcal{X}| = 2^{2^n}.$$

设 $\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2, \cdots, Z_r\}$ 为 $\mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, \cdots, X_n\}$ 的一个子空间, 那么, 每个 $Z_i \in \mathcal{X}$ 都是 $x_1, x_2, \cdots, x_n$ 的一个逻辑函数. 于是有

$$Z_i = G_i(X_1, X_2, \cdots, X_n), \quad i = 1, 2, \cdots, r.$$

设 G_i 的结构矩阵为 M_i, 那么, 在向量形式下有

$$z_i = M_i \ltimes_{j=1}^n x_j, \quad i=1,2,\cdots,r.$$

记 $z=\ltimes_{i=1}^r z_i$, $x=\ltimes_{i=1}^n x_i$, 于是有

$$z = M_{\mathcal{Z}} x, \tag{6.1.5}$$

这里

$$M_{\mathcal{Z}} := M_1 * M_2 * \cdots * M_e \in \mathcal{L}_{2^r \times 2^n},$$

称为子空间 $\mathcal{Z}$ 的结构矩阵. 容易证明, 子空间 $\mathcal{Z}$ 的性质完全由其结构矩阵决定.

6.1.2 状态空间的坐标变换

从现代控制理论中不难知道, 状态空间方法的强大生命力来自坐标变换, 因为适当的坐标能刻画系统特殊性质的各种子空间. 因此, 对于逻辑动态系统, 状态空间的坐标变换同样十分重要.

定义 6.1.2 设 $\mathcal{Z}=\mathcal{F}_\ell\{Z_1,Z_2,\cdots,Z_n\}\subset\mathcal{X}$. 为了记号的方便, 我们也将 $Z=(Z_1,Z_2,\cdots,Z_n)^{\mathrm{T}}$ 当作一个列向量. 由 $X=(X_1,X_2,\cdots,X_n)^{\mathrm{T}}\mapsto Z=(Z_1,Z_2\cdots,Z_n)^{\mathrm{T}}$ 所定义的映射 $G:\mathcal{D}^n\to\mathcal{D}^n$ 称为一个逻辑坐标变换, 如果 G 是一一且映上的.

下面考虑 G 的具体逻辑表达式:

$$G:\begin{cases} Z_1 = G_1(X_1,X_2,\cdots,X_n),\\ Z_2 = G_2(X_1,X_2,\cdots,X_n),\\ \qquad\cdots\cdots\\ Z_n = G_n(X_1,X_2,\cdots,X_n). \end{cases} \tag{6.1.6}$$

利用逻辑的向量表示, 记 $x_i=\vec{X}_i$, $z_i=\vec{Z}_i$, $i=1,2,\cdots,n$, $x=\ltimes_{i=1}^n x_i$, $z=\ltimes_{i=1}^n z_i$, 并设 G 的结构矩阵为 M_G. 于是可得到 G 的代数表达式

$$z = M_G x, \tag{6.1.7}$$

这里 M_i 为 G_i 的结构矩阵, $i=1,2,\cdots,n$, $M_G=M_1*M_2*\cdots*M_n\in\mathcal{L}_{2^n\times 2^n}$. 于是, 显然有如下结论.

定理 6.1.1 G 是一个坐标变换, 当且仅当, 其结构矩阵 M_G 非奇异.

注 6.1.1 如果 $T\in\mathcal{L}_{s\times s}$ 并且非奇异, 则 T 为一正交矩阵. 因此, 如果 G 是一个逻辑坐标变换, 那么, M_G 就是一个正交矩阵, 于是式 (6.1.7) 等价于

$$x = M_G^{\mathrm{T}} z. \tag{6.1.8}$$

下面给出一个逻辑坐标变换的具体例子.

例 6.1.1 设

$$\begin{cases} Z_1 = \neg X_2, \\ Z_2 = X_1 \leftrightarrow X_2, \\ Z_3 = \neg X_3, \end{cases} \tag{6.1.9}$$

那么

$$\begin{aligned} z &= z_1 z_2 z_3 \\ &= M_n x_2 M_e x_1 x_2 M_n x_3 \\ &= M_n (I_2 \otimes M_e) W_{[2,2]} x_1 x_2^2 M_n x_3 \\ &= M_n (I_2 \otimes M_e) W_{[2,2]} (I_2 \otimes \mathrm{PR}_2) x_1 x_2 M_n x_3 \\ &= M_n (I_2 \otimes M_e) W_{[2,2]} (I_2 \otimes \mathrm{PR}_2)(I_4 \otimes M_n) x_1 x_2 x_3 \\ &:= Tx, \end{aligned} \tag{6.1.10}$$

这里 $T \in \mathcal{L}_{8\times 8}$ 为

$$\begin{aligned} T &= M_n (I_2 \otimes M_e) W_{[2]} (I_2 \otimes R_2^P)(I_4 \otimes M_n) \\ &= \delta_8[6,5,4,3,8,7,2,1]. \end{aligned} \tag{6.1.11}$$

容易检验 T 是非奇异的, 因此, 式 (6.1.9) 是一个逻辑坐标变换. 其逆变换为

$$\begin{cases} X_1 = Z_1 \bar{\vee} Z_2, \\ X_2 = \neg Z_1, \\ X_3 = \neg Z_3. \end{cases} \tag{6.1.12}$$

下面考虑一般布尔网络的逻辑坐标变换.

给定一个布尔网络, 设其代数状态空间表达式为

$$x(t+1) = Mx(t), \quad x \in \Delta_{2^n}. \tag{6.1.13}$$

令 $z = Tx : \Delta_{2^n} \to \Delta_{2^n}$ 为一逻辑坐标变换, 那么

$$z(t+1) = Tx(t+1) = TMx(t) = TLT^{-1}z(t).$$

即, 布尔网络 (6.1.13) 在 z 坐标下的动态方程变为

$$z(t+1) = \tilde{M} z(t), \tag{6.1.14}$$

这里

$$\tilde{M} = TLT^{\mathrm{T}}.$$

下面考虑布尔控制网络, 设其代数状态空间表达式为

$$\begin{cases} x(t+1) = Mu(t)x(t), & x \in \Delta_{2^n},\ u \in \Delta_{2^m}, \\ y(t) = Hx(t), & y \in \Delta_{2^p}. \end{cases} \tag{6.1.15}$$

设 $z = Tx : \Delta_{2^n} \to \Delta_{2^n}$ 为一逻辑坐标变换. 类似于布尔网络, 布尔控制网络的动态方程 (6.1.15) 在新坐标下可表示为

$$\begin{cases} z(t+1) = \tilde{M}u(t)z(t), & z \in \Delta_{2^n},\ u \in \Delta_{2^m}, \\ y(t) = \tilde{H}z(t), & y \in \Delta_{2^p}, \end{cases} \tag{6.1.16}$$

这里

$$\begin{cases} \tilde{M} = TM(I_{2^m} \otimes T^{\mathrm{T}}), \\ \tilde{H} = HT^{\mathrm{T}}. \end{cases} \tag{6.1.17}$$

下面给出一个例子.

例 6.1.2 考虑如下布尔控制网络:

$$\begin{cases} X_1(t+1) = X_1(t) \to X_3(t), \\ X_2(t+1) = X_2(t) \leftrightarrow X_3(t), \\ X_3(t+1) = U(t) \wedge X_1(t), \end{cases} \tag{6.1.18}$$
$$Y(t) = X_1(t) \leftrightarrow X_2(t).$$

其代数状态空间表达式为

$$\begin{aligned} x_1(t+1) &= M_i x_1(t) x_3(t) = M_i(\mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}})u(t)x(t) := M_1 u(t)x(t), \\ x_2(t+1) &= M_e x_2(t) x_3(t) = M_e(\mathbf{1}_4^{\mathrm{T}} \otimes I_4)u(t)x(t) := M_2 u(t)x(t), \\ x_3(t+1) &= M_c u(t) x_1(t) = M_c(I_4 \otimes \mathbf{1}_4^{\mathrm{T}})u(t)x(t) := M_3 u(t)x(t), \end{aligned}$$

这里

$$\begin{aligned} M_1 &= M_i(\mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}}) \\ &= \delta_2[1,2,1,2,1,1,1,1,1,2,1,2,1,1,1,1], \\ M_2 &= M_e(\mathbf{1}_4^{\mathrm{T}} \otimes I_4) \end{aligned}$$

$$
\begin{aligned}
&= \delta_2[1,2,2,1,1,2,2,1,1,2,2,1,1,2,2,1],\\
M_3 &= M_c(I_4 \otimes \mathbf{1}_4^{\mathrm{T}})\\
&= \delta_2[1,1,1,1,2,2,2,2,2,2,2,2,2,2,2,2].\\
y(t) &= M_e x_1(t) x_2(t) = M_e(I_4 \otimes \mathbf{1}_2^{\mathrm{T}})x(t) := Hx(t).
\end{aligned}
$$

于是有

$$
\begin{cases}
x(t+1) = Mu(t)x(t),\\
y(t) = Hx(t),
\end{cases} \tag{6.1.19}
$$

这里

$$
\begin{aligned}
M &= M_1 * M_2 * M_3\\
&= \delta_8[1,7,3,5,2,4,4,2,2,8,4,6,2,4,4,2],\\
H &= M_e(I_4 \otimes \mathbf{1}_2^{\mathrm{T}})\\
&= \delta_2[1,1,2,2,2,2,1,1].
\end{aligned}
$$

下面考虑坐标变换 (6.1.9). 即 $z = Tx$, 这里

$$
T = \delta_8[6,5,4,3,8,7,2,1].
$$

利用公式 (6.1.17), 在 Z 坐标下式 (6.1.19) 变为

$$
\begin{cases}
z(t+1) = \tilde{M}u(t)z(t),\\
y(t) = \tilde{H}x(t),
\end{cases} \tag{6.1.20}
$$

这里

$$
\begin{aligned}
\tilde{M} &= TM(I_2 \otimes T^{\mathrm{T}})\\
&= \delta_8[5,3,8,4,2,6,3,5,5,3,7,3,1,5,3,5],\\
\tilde{H} &= HT^{\mathrm{T}}\\
&= \delta_2[1,1,2,2,1,1,2,2].
\end{aligned}
$$

6.1.3　正规子空间

定义 6.1.3　设

$$
\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2, \cdots, Z_r\} \subset \mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, \cdots, X_n\}
$$

为一逻辑子空间. $\mathcal{Z}$ 称为一个正规子空间, 如果存在 $\{Z_{r+1}, Z_{r+2}, \cdots, Z_n\} \subset \mathcal{X}$ 使得

$$
T: (X_1, X_2, \cdots, X_n) \mapsto (Z_1, Z_2, \cdots, Z_n)
$$

为一坐标变换.

讨论几个例子.

例 6.1.3 给定一个三维逻辑状态空间 $\mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, X_3\}$. 考察其子空间 $\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2\} \subset \mathcal{X}$.

(i) 设

$$\begin{cases} Z_1 = X_1 \leftrightarrow X_2, \\ Z_2 = X_2 \bar{\vee} X_3. \end{cases} \tag{6.1.21}$$

容易算出, 它们的代数表达式为

$$\begin{cases} z_1 = M_1 x_1 x_2 x_3, \\ z_2 = M_2 x_1 x_2 x_3, \end{cases} \tag{6.1.22}$$

这里

$$\begin{cases} M_1 = \delta_2[1,1,2,2,2,2,1,1], \\ M_2 = \delta_2[2,1,1,2,2,1,1,2]. \end{cases}$$

我们通过构造坐标变换来证明 $\mathcal{Z}$ 为一个正规子空间. 为此设

$$Z_3 = (X_1 \wedge (X_2 \leftrightarrow X_3)) \vee (\neg X_1 \wedge (X_2 \bar{\vee} X_3)).$$

在向量形式下, 它可表示为

$$z_3 = M_3 x_1 x_2 x_3, \tag{6.1.23}$$

这里

$$M_3 = \delta_2[1,2,2,1,2,1,1,2].$$

记 $z = z_1 z_2 z_3$ 及 $x = x_1 x_2 x_3$, 容易算得

$$z = M_1 * M_2 * M_3 x := Mx, \tag{6.1.24}$$

这里

$$M = \delta_8[3,2,6,7,8,5,1,4].$$

因为 $M \in \mathcal{L}_{8\times 8}$ 非奇异, z 是一组新坐标, 所以 $\mathcal{Z}$ 是一个正规子空间.

(ii) 设

$$\begin{cases} Z_1 = X_1 \to X_2, \\ Z_2 = X_2 \bar{\vee} X_3, \end{cases} \tag{6.1.25}$$

则它们的代数表达式为

$$\begin{cases} z_1 = M_1 x, \\ z_2 = M_2 x, \end{cases} \tag{6.1.26}$$

这里

$$\begin{cases} M_1 = \delta_2[1,1,2,2,1,1,1,1], \\ M_2 = \delta_2[2,1,1,2,2,1,1,2]. \end{cases}$$

设 $Z_3 \in \mathcal{X}$, 其代数表达式为

$$z_3 = \delta_2[r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8] := M_3 x,$$

于是有 $z = Mx$, 这里 $M = M_1 * M_2 * M_3$. 即

$$\mathrm{Col}_i(M) = \mathrm{Col}_i(M_1) \ltimes \mathrm{Col}_i(M_2) \ltimes \mathrm{Col}_i(M_3), \quad i = 1, 2, \cdots, 8.$$

现在

$$\mathrm{Col}_2(M) = \delta_2^1 \ltimes \delta_2^1 \ltimes \delta_2^{r_2},$$
$$\mathrm{Col}_6(M) = \delta_2^1 \ltimes \delta_2^1 \ltimes \delta_2^{r_6},$$
$$\mathrm{Col}_7(M) = \delta_2^1 \ltimes \delta_2^1 \ltimes \delta_2^{r_7}.$$

因为 $r_i \in \{1,2\}$, r_2, r_6, r_7 中必有两个相等. 故 M 是奇异的, 即 $\mathcal{Z}$ 不是一个正规子空间.

给定一组逻辑函数 Z_i, $i = 1, 2, \cdots, r$, 设其逻辑表达式为

$$Z_i = G_i(X_1, X_2, \cdots, X_n), \quad i = 1, 2, \cdots, r, \tag{6.1.27}$$

设子空间 $\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2, \cdots, Z_r\}$. 我们想知道, 何时 $\mathcal{Z}$ 为一个正规子空间? 如果 $\mathcal{Z}$ 为一个 k 维正规子空间, 则 $\{Z_1, Z_2, \cdots, Z_r\}$ 称为它的基底. 设 $\mathcal{Z}$ 的结构矩阵为 $M_{\mathcal{Z}}$, 即

$$z = M_{\mathcal{Z}} x,$$

这里 $M_{\mathcal{Z}} \in \mathcal{L}_{2^k \times 2^n}$.

下面的定理给出正规子空间的一个完整刻画.

定理 6.1.2 *考察网络* (6.1.1) (*或控制网络* (6.1.2)). *设* $\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2, \cdots, Z_r\}$, *其结构矩阵为* $M_{\mathcal{Z}} \in \mathcal{L}_{2^r \times 2^n}$. $\mathcal{Z}$ *为一个* r *维正规子空间, 且* $\{Z_1, Z_2, \cdots, Z_r\}$ *为其基底, 当且仅当,*

$$\sum_{i=1}^{2^n} \mathrm{Col}_i(M_{\mathcal{Z}}) = 2^{n-r} \mathbf{1}_{2^r}. \tag{6.1.28}$$

换言之, $M_{\mathcal{Z}}$ 具有 2^r 组互不相同的列: $\delta_{2^r}^i$, $i=1,2,\cdots,2^r$, 而每一组的个数均为 2^{n-r}.

证明　(充分性) 记 $z^r=\ltimes_{i=1}^r z_i$. $Z^r=\{Z_1,Z_2,\cdots,Z_r\}$, 并记

$$M^r:=M_{\mathcal{Z}}.$$

由 Z^r 定义

$$X_i^r:=\{x\mid M_{\mathcal{Z}}x=\delta_{2^r}^i\},\quad i=1,2,\cdots,2^r.$$

因为 M^r 满足 (6.1.28),

$$|X_i^r|=2^{n-r},\quad \forall i.$$

定义 Z_{r+1} 如下: 它在 X_i^r 中任选一半 x, 令 $z_{k+1}(x)=\delta_2^1$, 在另一半 x 上令 $z_{k+1}(x)=\delta_2^2$. 对所有的 i 都选完以后, z_{r+1} 就被唯一定义好了. 注意到, 元素 $\{Z_i\}$ 的排序与 (6.1.28) 无关, 如果 $z^r=\delta_{2^r}^i$, 则不妨设

$$z^{r+1}\sim z_{r+1}\ltimes z^r=\begin{cases}\begin{bmatrix}\delta_{2^r}^i\\ \mathbf{0}_{2^r}\end{bmatrix}, & z_{r+1}=\delta_2^1,\\ \begin{bmatrix}\mathbf{0}_{2^r}\\ \delta_{2^r}^i\end{bmatrix}, & z_{r+1}=\delta_2^2,\end{cases}$$

这里 $\sim$ 表示乘积因子的顺序换了. 记

$$z^{r+1}=M^{r+1}x,$$

由构造可知, M^{r+1} 中为 δ_{r+1}^i 的列有 2^{n-r-1} 个, 因此, 对 $\{Z_1,Z_2,\cdots,Z_{r+1}\}$, M^{r+1} 仍然满足 (6.1.28). 类此, 可以继续逐次增加 $Z_{r+2},\cdots,Z_n$, 使相应的 M^n 依然满足 (6.1.28). 即, 每个 $\delta_{2^n}^i$ 都出现且仅出现一次, 这表明 M^n 是非奇异的, 于是, $\{Z_1,Z_2,\cdots,Z_n\}$ 构成一新坐标变量.

(必要性) 首先证明一个断言: 如果 $\{Z_i\mid i=1,2,\cdots,r\}$ 张成的子空间的结构矩阵满足 (6.1.28), 那么, 由它的子集张成的子空间的结构矩阵也满足 (6.1.28). 注意到 Z_i 的排列顺序不会影响 (6.1.28), 我们只要证明, 如果 $\{Z_i\mid i=1,2,\cdots,r\}$ 张成的子空间的结构矩阵 M^r 满足 (6.1.28), 那么, $\{Z_i\mid i=2,\cdots,k\}$ 张成的子空间的结构矩阵 $\bar{M}^{r-1}\sim M^{r-1}$ 也满足 (6.1.28). 因为

$$z^r=\ltimes_{i=1}^r z_i=M^r x,$$

两边左乘 $\mathbf{1}_2^{\mathrm{T}}$ 则得

$$\bar{z}^{r-1}=\ltimes_{i=2}^r z_i=\mathbf{1}_2^{\mathrm{T}}M^r x:=\bar{M}^{r-1}x.$$

因此, 只要证明 $\bar{M}^{r-1} = \mathbf{1}_2^{\mathrm{T}} M_k$ 满足 (6.1.28) 即可. 记

$$M^r = \begin{bmatrix} M_1^r \\ M_2^r \end{bmatrix},$$

这里 $M_i^r \in \mathcal{B}_{2^{r-1}\times 2^n}$, $i = 1, 2$. 于是

$$\bar{M}^{r-1} = \mathbf{1}_2^{\mathrm{T}} M^r = M_r^1 + M_r^2. \tag{6.1.29}$$

根据 (6.1.29), M^r 中的列到 $\bar{M}^{r-1}$ 中的列的映射可如下实现:

$$\begin{bmatrix} \delta_{2^{r-1}}^i \\ \mathbf{0}_{2^{r-1}} \end{bmatrix}, \begin{bmatrix} \mathbf{0}_{2^{r-1}} \\ \delta_{2^{r-1}}^i \end{bmatrix} \mapsto \delta_{2^{r-1}}^i,$$

注意到 M_r 中上述两类向量 (即这样的列) 各有 2^{n-r} 个, 于是 $\bar{M}^{r-1}$ 中形如 $\delta_{2^{r-1}}^i$, $i = 1, 2, \cdots, 2^{r-1}$ 的列各有 2^{n-r+1} 个. 显然, (6.1.28) 成立. 这就证明了断言.

考察 $\{z_1, z_2, \cdots, z_n\}$, 作为坐标变量 (的向量表示), 相应的 M^n 显然满足 (6.1.28), 因为这时, 每个 $\delta_{2^n}^i$ 型的列均出现且仅出现一次. 利用上述断言, 子集 $\{z_1, z_2, \cdots, z_r\}$ 所对应的 M^r 也满足 (6.1.28). □

实际上, 我们也可以直接由 G_i, $i = 1, 2, \cdots, r$ 的结构矩阵来判定正规子空间. 设 G_i 的结构矩阵为

$$M_i = \delta_2[j_{i,1}, j_{i,2}, \cdots, j_{i,2^n}], \quad i = 1, 2, \cdots, r.$$

定义其示性矩阵为

$$J(M_1, M_2, \cdots, M_r) := \begin{bmatrix} j_{1,1} & j_{1,2} & \cdots & j_{1,2^n} \\ j_{2,1} & j_{2,2} & \cdots & j_{2,2^n} \\ \vdots & \vdots & & \vdots \\ j_{r,1} & j_{r,2} & \cdots & j_{r,2^n} \end{bmatrix}. \tag{6.1.30}$$

那么, 不难得到 $\mathcal{Z}$ 与 J 之间具有如下关系.

推论 6.1.1　*$\mathcal{Z}$ 为一个 r 维正规子空间, 且 $\{Z_1, Z_2, \cdots, Z_r\}$ 为其基底, 当且仅当, J 具有 2^r 组互不相同的列, 而每一组的个数均为 2^{n-r}.*

下面通过几个例子说明怎样从正规子空间构造新坐标.

例 6.1.4　(i) 回忆例 6.1.3 的第一个例子. 由式 (6.1.22) 可得

$$J = \begin{bmatrix} 1 & 1 & 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 1 & 1 & 2 & 2 & 1 & 1 & 2 \end{bmatrix}.$$

它显然满足推论 6.1.1 的要求. 要构造新坐标, 只要补上的坐标变量使扩充的 G 的各列都不相同即可. 例如, $[1,2,2,1,2,1,1,2]$ 显然满足要求. 其实, 这就是式 (6.1.23). 还有许多选择, 对这个例子, 只要 $j_1 \neq j_8$, $j_2 \neq j_7$, $r_3 \neq r_6$, $r_4 \neq r_5$ 就可以了. 所以, 可以选 $[j_1, j_2, j_3, j_4, 3-j_4, 3-j_3, 3-j_2, 3-j_1]$. 例如 $[2,1,1,2,1,2,2,1]$, 等等.

(ii) 设 $\mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, X_3, X_4\}$, $\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2\}$, 这里

$$
\begin{aligned}
z_1 &= \delta_2[2,1,1,1,1,2,2,1,2,2,1,1,2,1,2,2]x,\\
z_2 &= \delta_2[2,1,1,2,1,1,1,2,2,2,2,1,2,2,1,1]x.
\end{aligned}
$$

容易看出, $\mathcal{Z}$ 是一个正规子空间. 要利用 $\{z_1, z_2\}$ 构造新坐标, 只要选择 $\{z_3, z_4\}$ 使示性矩阵 $J(z_1, z_2, z_3, z_4)$ 的各列均不相同即可. 以下为一种选法:

$$
\begin{aligned}
z_3 &= \delta_2[1,1,2,2,1,2,1,1,1,2,2,2,2,1,1,2]x,\\
z_4 &= \delta_2[1,2,2,1,1,2,1,1,2,2,2,1,1,2,2,1]x.
\end{aligned}
$$

6.2 不变子空间

6.2.1 布尔网络的不变子空间

定义 6.2.1 给定一个布尔网络

$$
\begin{cases}
X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t)),\\
X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t)),\\
\qquad\qquad \cdots\cdots\\
X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t)).
\end{cases}
\tag{6.2.1}
$$

$\mathcal{Z}^1 = \mathcal{F}_\ell\{Z^1\} = \mathcal{F}_\ell\{Z_1^1, Z_2^1, \cdots, Z_r^1\}$ 称为系统 (6.2.1) 的不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且, 在 Z 坐标下系统可表示为

$$
\begin{cases}
Z^1(t+1) = \tilde{F}^1(Z^1(t)), \quad Z^1(t) \in \mathcal{D}^r,\\
Z^2(t+1) = \tilde{F}^2(Z(t)), \quad Z^2(t) \in \mathcal{D}^{n-r}.
\end{cases}
\tag{6.2.2}
$$

设式 (6.2.1) 的代数状态空间表示为

$$
x(t+1) = Mx(t),
\tag{6.2.3}
$$

则有如下结论.

定理 6.2.1　考察系统 (6.2.1) 及其代数形式 (6.2.3). 设 $\mathcal{Z}^1 = \mathcal{F}_\ell\{Z_1^1, Z_2^1, \cdots, Z_r^1\}$ 为正规子空间, 其代数表达式为

$$z^1 = Qx, \tag{6.2.4}$$

这里 $z^1 = \ltimes_{i=1}^r z_i^1$, $Q \in \mathcal{L}_{2^r \times 2^n}$. 那么, $\mathcal{Z}^1$ 为系统 (6.2.1) 的不变子空间, 当且仅当, 存在 $H \in \mathcal{L}_{2^r \times 2^r}$ 使得

$$QM = HQ. \tag{6.2.5}$$

证明　(充分性) 因为 $\mathcal{Z}^1$ 是一个正规子空间, 则存在 $Z^2 = \{Z_{r+1}^2, Z_{r+2}^2, \cdots, Z_n^2\}$, 使得 $Z = \{Z^1, Z^2\}$ 为一新坐标. 于是,

$$z^1(t+1) = Qx(t+1) = QMx(t) = HQx(t) = Hz^1(t). \tag{6.2.6}$$

(6.2.6) 表明在 Z 坐标下系统具有 (6.2.2) 的形式.

(必要性) 设系统 (6.2.1) 在 Z 坐标下具有 (6.2.2) 形式. 并设 $\tilde{M}_1 \in \mathcal{L}_{2^k \times 2^k}$ 为 $\tilde{F}_1$ 的结构矩阵. 因此,

$$z^1(t+1) = \tilde{M}_1 z^1(t) = \tilde{M}_1 Qx(t).$$

另一方面,

$$z^1(t+1) = Qx(t+1) = QMx(t).$$

因为 $x(t)$ 任意, 则有

$$QM = \tilde{M}_1 Q.$$

令 $H = \tilde{M}_1$, 则得 (6.2.5). □

注 6.2.1　根据定理 6.2.1, 检验一个正规子空间是否为不变子空间, 关键是看何时方程 (6.2.5) 有解 H. 因为 $\mathcal{Z}^1$ 为正规子空间, 其结构矩阵 Q 必定行满秩. 因此, H 如果是解, 则必等于以下的 H^*:

$$H^* := QMQ^{\mathrm{T}}(QQ^{\mathrm{T}})^{-1}. \tag{6.2.7}$$

因此, 检验方程 (6.2.5) 是否有解 H 可依以下步骤进行.

第一步: 利用公式 (6.2.7) 计算 H^*. 如果 H^* 不是逻辑矩阵, 则 (6.2.5) 无所要求的解, 则 $\mathcal{Z}^1$ 不是不变子空间.

第二步: 如果 H^* 是逻辑矩阵, 则将其代入 (6.2.5) 以检验 H^* 是否为真解. 只有当 H^* 为真解时 $\mathcal{Z}^1$ 才是不变子空间.

下面给出一个例子.

例 6.2.1 考察如下系统:

$$\begin{cases} X_1(t+1) = (X_1(t) \wedge X_2(t) \wedge \neg X_4(t)) \vee (\neg X_1(t) \wedge X_2(t)), \\ X_2(t+1) = X_2(t) \vee (X_3(t) \leftrightarrow X_4(t)), \\ X_3(t+1) = (X_1(t) \wedge \neg X_4(t)) \vee (\neg X_1(t) \wedge X_2(t)) \\ \qquad\qquad\quad \vee(\neg X_1(t) \wedge \neg X_2(t) \wedge X_4(t)), \\ X_4(t+1) = X_1(t) \wedge \neg X_2(t) \wedge X_4(t). \end{cases} \tag{6.2.8}$$

易知, 其代数表示为

$$x(t+1) = Mx(t), \tag{6.2.9}$$

这里

$$M = \delta_{16}[11, 1, 11, 1, 11, 13, 15, 9, 1, 2, 1, 2, 9, 15, 13, 11].$$

设 $\mathcal{Z} = \mathcal{F}_\ell\{Z_1, Z_2, Z_3\}$, 这里

$$\begin{cases} Z_1 = X_1 \bar{\vee} X_4, \\ Z_2 = \neg X_2, \\ Z_3 = X_3 \leftrightarrow \neg X_4. \end{cases} \tag{6.2.10}$$

记 $x = \ltimes_{i=1}^4 x_i$, $z = \ltimes_{i=1}^3 z_i$, 则有

$$z = Qx,$$

这里

$$Q = \delta_8[8, 3, 7, 4, 6, 1, 5, 2, 4, 7, 3, 8, 2, 5, 1, 6].$$

利用 (6.2.7), 可得

$$H^* = \delta_8[2, 4, 8, 8, 1, 3, 3, 3].$$

它满足式 (6.2.5). 因此 $\mathcal{Z}$ 是系统 (6.2.8) 的不变子空间.

实际上, 可选 $Z_4 = X_4$, 使得

$$\begin{cases} Z_1 = X_1 \bar{\vee} X_4, \\ Z_2 = \neg X_2, \\ Z_3 = X_3 \leftrightarrow \neg X_4, \\ Z_4 = X_4 \end{cases} \tag{6.2.11}$$

为一坐标变换. 而在这个新坐标下, 系统 (6.2.8) 变为

$$\begin{cases} Z_1(t+1) = Z_1(t) \to Z_2(t), \\ Z_2(t+1) = Z_2(t) \wedge Z_3(t), \\ Z_3(t+1) = \neg Z_1(t), \\ Z_4(t+1) = Z_1(t) \vee Z_2(t) \vee Z_4(t). \end{cases} \tag{6.2.12}$$

式 (6.2.12) 显然具有如式 (6.2.2) 所示的标准形式.

6.2.2　布尔控制网络的不变子空间

定义 6.2.2　给定一个布尔控制网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)). \end{cases} \tag{6.2.13}$$

(i) $\mathcal{Z}^1 = \mathcal{F}_\ell\{Z^1\} = \mathcal{F}_\ell\{Z_1^1, Z_2^1, \cdots, Z_r^1\}$ 称为系统 (6.2.13) 的 X 不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且, 在 Z 坐标下系统可表示为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t)), & Z^1(t) \in \mathcal{D}^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t), U(t)), & Z^2(t) \in \mathcal{D}^{n-r}. \end{cases} \tag{6.2.14}$$

(ii) $\mathcal{Z}^1$ 称为系统 (6.2.13) 的 UX 不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且, 在 Z 坐标下系统可表示为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t), U(t)), & Z^1(t) \in \mathcal{D}^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t), U(t)), & Z^2(t) \in \mathcal{D}^{n-r}. \end{cases} \tag{6.2.15}$$

(iii) $\mathcal{Z}^1$ 称为系统 (6.2.13) 的控制 X 不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且存在一个定常控制 $U(t) = U$, 使在这个控制下 $\mathcal{Z}^1$ 成为一不变子空间, 即系统变为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t)), & Z^1(t) \in \mathcal{D}^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t)), & Z^2(t) \in \mathcal{D}^{n-r}. \end{cases} \tag{6.2.16}$$

设式 (6.2.13) 的代数状态空间表示为

$$x(t+1) = Mu(t)x(t). \tag{6.2.17}$$

那么, 有如下结论.

命题 6.2.1 设正规子空间 $\mathcal{Z}^1 \subset \mathcal{X}$ 的向量表达为

$$z^1 = Qx.$$

(i) $\mathcal{Z}^1$ 为 X 不变子空间, 当且仅当, 存在 $H \in \mathcal{L}_{2^r \times 2^r}$ 使得

$$QM = H(\mathbf{1}_{2^m}^{\mathrm{T}} \otimes Q). \tag{6.2.18}$$

(ii) $\mathcal{Z}^1$ 为 UX 不变子空间, 当且仅当, 存在 $H \in \mathcal{L}_{2^r \times 2^{m+r}}$ 使得

$$QM = HQ. \tag{6.2.19}$$

(iii) 将 M 均分为 2^m 块, 即

$$M = [M_1, M_2, \cdots, M_{2^m}],$$

这里 $M_i = M\delta_{2^m}^i \in \mathcal{L}_{2^n \times 2^n}$, $i = 1, 2, \cdots, 2^m$. $\mathcal{Z}^1$ 为控制 X 不变子空间, 当且仅当, 存在 i 使得 $\mathcal{Z}^1$ 为

$$x(t+1) = M_i x(t)$$

的不变子空间.

证明 记 (6.2.13) 的代数状态空间表达为

$$x(t+1) = Mu(t)x(t). \tag{6.2.20}$$

设 $u = \delta_{2^m}^i$, 则对这个固定的 i, (6.2.20) 变为

$$x(t+1) = M_i x(t), \tag{6.2.21}$$

这里 $M_i = M\delta_{2^n}^i$. (6.2.15) 变为

$$\begin{cases} z^1(t+1) = \tilde{M}_i^1 z^1(t), \\ z^2(t+1) = \tilde{M}_i^2 z(t). \end{cases} \tag{6.2.22}$$

利用定理 6.2.1, (6.2.21) 能转化为 (6.2.22), 当且仅当, 存在 $H_i \in \mathcal{L}_{2^r \times 2^r}$, 使

$$QM_i = H_i Q, \quad i = 1, 2, \cdots, 2^m. \tag{6.2.23}$$

记 $H := [H_1, H_2, \cdots, H_{2^r}]$, 则得 $QM = HQ$, 这就是 (6.2.19).

现在考察 (6.2.14), 实际上, (6.2.23) 依然成立, 只是现在

$$H_1 = H_2 = \cdots = H_{2^m} := H = M_{\tilde{F}^1},$$

于是立得 (6.2.18)

最后考虑控制 X 不变子空间的情况. 对某定常控制 $u(t) = \delta_{2^m}^i$, 系统变为 (6.2.21). 类似上面的讨论即得结论. □

下面给出一个例子.

例 6.2.2 考察下列系统

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), X_3(t), U(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), X_3(t), U(t)), \\ X_2(t+1) = F_3(X_1(t), X_2(t), X_3(t), U(t)), \end{cases} \tag{6.2.24}$$

这里

$$\begin{aligned} &F_1(X_1(t), X_2(t), X_3(t), U(t)) \\ =& \{U(t) \wedge [(X_1(t) \wedge X_2(t)) \vee (X_1(t) \wedge \neg X_2(t) \wedge \neg X_3(t)) \\ &\vee (\neg X_1(t) \wedge X_2(t) \wedge X_3(t))]\} \\ &\vee \{\neg U(t) \wedge [(X_1(t) \wedge \neg X_2(t) \wedge X_3(t)) \\ &\vee (\neg X_1(t) \wedge X_2(t) \wedge \neg X_3(t)) \\ &\vee (\neg X_1(t) \wedge \neg X_2(t))]\}, \\ F_2&(X_1(t), X_2(t), X_3(t), U(t)) = (U(t) \wedge X_1(t)) \vee (\neg U(t) \wedge \neg X_1(t)), \\ &F_3(X_1(t), X_2(t), X_3(t), U(t)) \\ =& (X_1(t) \wedge X_2(t) \wedge X_3(t)) \vee (X_1(t) \wedge \neg X_2(t)) \\ &\vee (\neg X_1(t) \wedge X_2(t)) \vee (\neg X_1(t) \wedge \neg X_2(t) \wedge X_3(t)). \end{aligned}$$

它的代数状态空间表达式为

$$x(t+1) = M_x u(t) x(t), \tag{6.2.25}$$

这里

$$M_x = \delta_8[1, 2, 5, 1, 3, 7, 7, 8, 7, 8, 3, 7, 5, 1, 1, 2].$$

考察一个子空间 $\mathcal{Z}^1 = \mathcal{F}_\ell(Z_1, Z_2)$, 这里

$$\begin{aligned} Z_1 &= X_1 \bar{\vee} X_2, \\ Z_2 &= X_3. \end{aligned} \tag{6.2.26}$$

(6.2.26) 的代数表示式为

$$z^1(t) = Qx(t),$$

这里

$$Q = \delta_4[3,4,1,2,1,2,3,4].$$

根据定理 6.1.2, 容易验证 $\mathcal{Z}^1$ 是个正规子空间, 并且有 $\mathrm{Row}(QM) \subset \mathrm{span}_{\mathcal{B}}(Q)$. 从而找出

$$H = \delta_4[1,3,3,4].$$

容易检验

$$QM_x = H(\mathbf{1}_2^{\mathrm{T}} \otimes Q).$$

因此, $\mathcal{Z}^1$ 就是一个 X 不变子空间. 例如, 令 $Z_3(t) = \neg X_2(t)$, 则得

$$z(t) = Tx(t) = \delta_8[6,8,1,3,2,4,5,7]x(t).$$

显然, T 是一个坐标变换. 于是

$$\begin{aligned} z(t+1) &= Tx(t+1) = TM_x u(t)x(t) \\ &= TM_x u(t)T^{\mathrm{T}}z(t) = TM_x(I_2 \otimes T^{\mathrm{T}})u(t)z(t) \\ &:= M_z u(t)z(t), \end{aligned}$$

这里

$$\begin{aligned} M_z &= TM_x(I_2 \otimes T^{\mathrm{T}}) \\ &= \delta_8[2,1,6,5,6,5,7,8,1,2,5,6,6,5,8,7]. \end{aligned}$$

返回逻辑系统形式, 则得

$$\begin{cases} Z_1(t+1) = Z_1(t) \wedge Z_2(t), \\ Z_2(t+1) = Z_1(t) \vee Z_2(t), \\ Z_3(t+1) = (Z_1(t) \leftrightarrow Z_3(t)) \bar{\vee} U(t). \end{cases} \tag{6.2.27}$$

根据定义, $\mathcal{Z}^1 = \mathcal{F}_\ell(Z_1, Z_2)$ 是 X 不变子空间.

注 6.2.2 (i) 用动态方程的方法定义不变子空间有清晰的物理意义: 设 $\mathcal{Z}^1$ 为不变子空间, 初值 $X_0 \in \mathcal{Z}^1$, 那么, 这条轨线就永远属于这个不变子空间.

(ii) 对于控制网络, 后面将看到, 三种不同的不变子空间各有自己的应用. 它们之间有如下关系:

$$X \text{ 不变子空间 } \Rightarrow UX \text{ 不变子空间 } \Rightarrow \text{控制 } X \text{ 不变子空间}.$$

6.3 布尔网络的标准型

本节讨论布尔网络的一个标准型, 它与布尔网络的拓扑结构是密切相关的. 本节的主要内容来自文献 [98].

6.3.1 标准型的结构

先介绍两类矩阵.

定义 6.3.1 (i) 一个逻辑矩阵 $C_k \in \mathcal{L}_{k\times k}$ 称为一个 k 阶循环矩阵 (cyclic matrix), 如果

$$C_k = \delta_k[2, 3, \cdots, k, 1]. \tag{6.3.1}$$

(ii) 一个方阵 N_k 称为幂零矩阵 (nilpotent matrix), 如果存在一个 s 使得 $N_k^s = 0$.

定义 6.3.2 考察布尔网络 (6.2.1). 如果存在一个坐标 Z, 使在此坐标上网络 (6.2.1) 的代数状态空间表示为

$$z(t+1) = \tilde{M} z(t), \tag{6.3.2}$$

这里

$$\tilde{M} = \operatorname{diag}\left(\tilde{M}_1, \tilde{M}_2, \cdots, \tilde{M}_k\right), \tag{6.3.3}$$

并且

$$\tilde{M}_j = \begin{bmatrix} C_j & D_j \\ 0 & N_j \end{bmatrix}, \quad j = 1, \cdots, k, \tag{6.3.4}$$

这里, C_j 是一个 n_j 阶循环矩阵, N_j 是 m_j 阶幂零矩阵. 那么, (6.3.2)—(6.3.4) 称为布尔网络 (6.2.1) 的标准型.

在证明一个布尔网络的标准型的存在性和唯一性之前, 先看看标准型的物理意义. 记

$$r_j := \begin{cases} 0, & j = 1, \\ \displaystyle\sum_{i=1}^{j-1} (n_i + m_i), & j = 2, \cdots, k. \end{cases}$$

那么, 我们有以下结果.

命题 6.3.1 如果布尔网络 (6.2.1) 有形如 (6.3.2)—(6.3.4) 的标准型, 那么,

(i) (6.2.1) 有 k 个吸引子, 为

$$\mathcal{C}_j := Z_1^j \to Z_2^j \to \cdots \to Z_{n_j}^j \to Z_1^j, \quad j = 1, \cdots, k, \tag{6.3.5}$$

这里

$$z_i^j = \vec{Z}_i^j = \delta_{2^n}^{r_j+i}, \quad i = 1, \cdots, n_j, \quad j = 1, \cdots, k. \tag{6.3.6}$$

(ii) 吸引子 $\mathcal{C}_j$ 的吸引域是

$$\mathcal{B}_j := \left\{B_1^j, B_2^j, \cdots, B_{m_j}^j\right\}, \quad j = 1, \cdots, k, \tag{6.3.7}$$

这里

$$b_i^j = \vec{B}_i^j = \delta_{2^n}^{r_j+n_j+i}, \quad i = 1, \cdots, m_j, \quad j = 1, \cdots, k. \tag{6.3.8}$$

证明 (i) 因为 $\mathrm{span}\{C_j\}$ 是 $\tilde{M}$ 不变的, 利用 C_j, 直接计算可得

$$\tilde{M} z_i^j = \begin{cases} z_1^j, & i = n_j, \\ z_{i+1}^j, & 1 \leqslant i \leqslant n_j - 1. \end{cases}$$

(ii) 因为 $\mathrm{span}\{\tilde{M}_j\}$ 是 $\tilde{M}$ 不变的, 我们有

$$(\tilde{M})^s = \mathrm{diag}\left(\tilde{M}_1^s, \tilde{M}_2^s, \cdots, \tilde{M}_k^s\right).$$

因为 N_{m_j} 是幂零的, 当 s 足够大时

$$\tilde{M}_j^s = \begin{bmatrix} C_{n_j}^s & * \\ 0 & 0 \end{bmatrix},$$

也就是说,

$$\tilde{M}^s b_i^j \in C_j, \quad i = 1, \cdots, m_j, \quad j = 1, \cdots, k. \qquad \square$$

命题 6.3.1 表明了标准型的物理意义. 实际上, 它准确刻画了布尔网络的拓扑结构. 下面证明, 每个布尔网络都有一个标准型, 而且, 在等价意义下标准型是唯一的.

定理 6.3.1 给定布尔网络 (6.2.1). 则存在一个坐标变换 $z = Tx$, 使得在坐标 z 下布尔网络 (6.2.1) 具有标准型 (6.3.2)—(6.3.4). 并且, 任何两个标准型在容许以下变换下是唯一的. (i) 闭环 C_j 内元素的循环变换; (ii) B_j 内元素的置换; 以及 (iii) 各块 $\{\tilde{M}_j\}$ 之间的置换.

注 6.3.1　1) 所谓 "在容许以下变换下是唯一的" 指的是: 任何一个标准型均可通过 (i)—(iii) 的变换变为另一个标准型. 由于这些变换不会改变标准型的基本结构, 不妨将它们看作一个 (即等价的). 因此, 在这个意义下标准型是唯一的.

2) 实际上, 每个块 $\tilde{M}_j$ 对应一个吸引子和它的吸引域, 因此, 每个块除内部点的顺序外是唯一的. 而块与块之间当然可以有不同排列顺序.

3) 这里不证明定理 6.3.1 . 实际上, 6.3.2 小节的算法给出了构造性证明.

6.3.2　标准型的算法

考察布尔网络 (6.2.1), 设其代数状态空间表示 (6.2.3) 已知, 即结构矩阵 M 已知. 回忆第 3 章, 定理 3.1.1 和定理 3.1.2 说明长度为 s 的极限环个数可计算如下:

$$
\begin{cases}
N_1 = \mathrm{tr}(M), \\
N_s = \dfrac{\mathrm{tr}(M^s) - \sum\limits_{k\in\mathcal{P}(s)} kN_k}{s}, \quad 2 \leqslant s \leqslant 2^n,
\end{cases} \tag{6.3.9}
$$

这里 $\mathcal{P}(s)$ 是 s 的真因子集. 第 3 章也给出了极限环及其吸引域的算法. 利用它们可得到以下算法.

算法 6.3.1　第一步: 计算所有的极限环 (不动点视为长度为 1 的极限环), 记作

$$
C_i = \{\xi_1^i, \xi_2^i, \cdots, \xi_{n_i}^i\}, \quad i = 1, \cdots, k. \tag{6.3.10}
$$

第二步: 计算每一个极限环的吸引域, 记作

$$
B_i = \{\eta_1^i, \eta_2^i, \cdots, \eta_{m_i}^i\}, \quad i = 1, \cdots, k. \tag{6.3.11}
$$

第三步: 构造坐标变换矩阵

$$
\begin{aligned}
T = [&\xi_1^1, \xi_2^1, \cdots, \xi_{n_1}^1, \eta_1^1, \eta_2^1, \cdots, \eta_{m_1}^1, \\
&\xi_1^2, \xi_2^2, \cdots, \xi_{n_2}^2, \eta_1^2, \eta_2^2, \cdots, \eta_{m_2}^2, \\
&\cdots, \\
&\xi_1^k, \xi_2^k, \cdots, \xi_{n_k}^k, \eta_1^k, \eta_2^k, \cdots, \eta_{m_k}^k].
\end{aligned}
$$

第四步: 构造坐标变换

$$
z = T^{\mathrm{T}} x. \tag{6.3.12}
$$

命题 6.3.2 利用算法 6.3.1, 布尔网络 (6.2.1) 在 z 坐标下具有标准型.

证明 根据定义不难看出, 一个坐标变换矩阵是一个置换矩阵, 因此它所实现的是一个状态的重排. 我们只看第一个块, T 将 $\delta_{2^n}^1$ 变为 ξ_1^1, $\delta_{2^n}^2$ 变为 ξ_2^1, $\cdots$, $\delta_{2^n}^{n_1}$ 变为 $\xi_{n_1}^1$; $\delta_{2^n}^{n_1+1}$ 变为 η_1^1, $\cdots$, $\delta_{2^n}^{n_1+m_1}$ 变为 $\eta_{m_1}^1$. 那么, $T^{\mathrm{T}}=T^{-1}$ 就将 ξ_1^1 变为 $\delta_{2^n}^1$, ξ_2^1 变为 $\delta_{2^n}^2$, 等等. 于是, 在 z 坐标下, 转移矩阵 $\tilde{M}$ 就具有形如 (6.3.3) 的块对角结构. 再看每一块, 记

$$\tilde{M}_j=\begin{bmatrix}C_j & D_j\\ H_j & N_j\end{bmatrix}.$$

显然 C_j 对应第 j 个吸引子, 它有循环结构. 因为环上的点不会移动到环外, 故有 $H_j=0$. 又因为

$$\tilde{M}_j^s=\begin{bmatrix}C_j^s & *\\ 0 & N_j^s\end{bmatrix},$$

当 s 足够大时, 吸引域上所有的点都会进入吸引子, 故当 s 足够大时 $N_j^s=0$. □

下面用一个例子描述这个算法.

例 6.3.1 考察布尔网络

$$\begin{cases}X_1(t+1)=\neg X_1(t)\wedge X_3(t),\\ X_2(t+2)=[\neg X_2(t)\wedge X_3(t)]\vee\neg X_1(t),\\ X_3(t+1)=X_1(t)\wedge[X_2(t)\bar{\vee}X_3(t)].\end{cases}\tag{6.3.13}$$

其代数状态空间表达式为

$$x(t+1)=Mx(t),\tag{6.3.14}$$

这里,

$$M=\delta_8[8,7,5,8,2,6,2,6].$$

不难算出, 它有两个吸引子.

(i) 不动点:

$$\delta_8^6.$$

它的吸引域是

$$\delta_8^1\to\delta_8^8\to\delta_8^6;\quad \delta_8^4\to\delta_8^8\to\delta_8^6.$$

(ii) 极限环:

$$\delta_8^7\to\delta_8^2\to\delta_8^7.$$

它的吸引域是

$$\delta_8^3 \to \delta_8^5 \to \delta_8^2.$$

将它们依次排成矩阵形式, 则得

$$T = \delta_8[6,8,1,4,2,7,3,4].$$

令 $z = T^{\mathrm{T}} x$, 则在 z 坐标下有

$$\tilde{M} = \delta_8[1,3,1,3,6,5,8,6]. \tag{6.3.15}$$

显然 (6.3.15) 是标准型, 其中

$$C_1 = [1], \qquad D_1 = [0,1,0], \quad N_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix},$$

$$C_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad D_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad N_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}.$$

最后可得

$$z(t+1) = \tilde{M} z(t),$$

这里

$$\tilde{M} = \delta_8[1,3,1,3,6,5,8,6].$$

6.4 k 值逻辑系统的状态空间方法

考察一个混合值逻辑网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t)), \end{cases} \tag{6.4.1}$$

这里 $X_i(t) \in \mathcal{D}_{k_i}$, $i = 1,2,\cdots,n$.

或者一个混合值逻辑控制网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \end{cases} \tag{6.4.2}$$
$$Y_j = h_j(X_1(t), X_2(t), \cdots, X_n(t)), \quad j = 1, 2, \cdots, p,$$

这里, $X_i(t) \in \mathcal{D}_{k_i}$, $i = 1, 2, \cdots, n$, $U_\alpha(t) \in \mathcal{D}_{r_\alpha}$, $\alpha = 1, 2, \cdots, m$, $Y_j(t) \in \mathcal{D}_{s_j}$, $j = 1, 2, \cdots, p$.

本章前面讨论的关于布尔网络的状态空间、子空间以及坐标变换等结果, 均可平行地推广到混合值逻辑 (控制) 网络上来. 但是, 在处理混合值逻辑网络时, 要仔细处理不同维数的逻辑变量和逻辑函数, 带来许多繁琐而又非本质的技术细节, 本节只讨论 k 值逻辑. 即, 在上述 (6.4.1) 及 (6.4.2) 中假定

$$k_i = r_\alpha = s_j := k, \quad \forall i, \alpha, j. \tag{6.4.3}$$

6.4.1 k 值逻辑网络的状态空间与坐标变换

定义 6.4.1 考虑 k 值逻辑网络 (6.4.1) 或 k 值逻辑控制网络 (6.4.2).

(i) 其状态空间, 记作 $\mathcal{X}$, 定义为由 $X_1, X_2, \cdots, X_n$ 的所有 k 值逻辑函数组成的集合, 记作 $\mathcal{F}_k\{X_1, X_2, \cdots, X_n\}$, 即

$$\mathcal{X} = \mathcal{F}_k\{X_1, X_2, \cdots, X_n\}. \tag{6.4.4}$$

(ii) 设 $Z_1, Z_2, \cdots, Z_r \in \mathcal{X}$, 那么, 由 $Z_1, Z_2, \cdots, Z_r$ 生成的子空间, 称为 k 值逻辑网络的状态子空间, 记作 $\mathcal{Z}$, 指 $Z_1, Z_2, \cdots, Z_r$ 的 k 值逻辑函数集合, 即

$$\mathcal{Z} = \mathcal{F}_k\{Z_1, Z_2, \cdots, Z_r\}. \tag{6.4.5}$$

设 $\mathcal{Z} = \mathcal{F}_k\{Z_1, Z_2, \cdots, Z_r\}$ 为 $\mathcal{X} = \mathcal{F}_k\{X_1, X_2, \cdots, X_n\}$ 的一个子空间, 那么, 它有一个代数表示

$$z = M_{\mathcal{Z}} x, \tag{6.4.6}$$

这里 $M_{\mathcal{Z}} \in \mathcal{L}_{k^r \times k^n}$ 称为子空间 $\mathcal{Z}$ 的结构矩阵.

定义 6.4.2 设 $\mathcal{Z} = \mathcal{F}_k\{Z_1, Z_2, \cdots, Z_n\} \subset \mathcal{X}$. 由 $X = (X_1, X_2, \cdots, X_n)^{\mathrm{T}} \mapsto Z = (Z_1, Z_2 \cdots, Z_n)^{\mathrm{T}}$ 所定义的映射 $G : \mathcal{D}_k^n \to \mathcal{D}_k^n$ 称为一个逻辑坐标变换, 如果 G 是一一映上的.

记 G 的结构矩阵为 M_G, 即

$$z = M_G x, \tag{6.4.7}$$

那么, 显然有如下结论.

定理 6.4.1　G 是一个坐标变换, 当且仅当, 其结构矩阵 M_G 非奇异.

k 值逻辑网络有与布尔网络相同的标准型.

定理 6.4.2　考察 k 值逻辑网络 (6.4.1). 存在一个坐标 Z, 使在此坐标下网络 (6.4.1) 的代数状态空间表达式为

$$z(t+1) = \tilde{M} z(t), \tag{6.4.8}$$

这里, 转移矩阵 $\tilde{M}$ 具有 (6.3.3)—(6.3.4) 的标准形式.

6.4.2　k 值逻辑网络的正规子空间与不变子空间

定义 6.4.3　设

$$\mathcal{Z} = \mathcal{F}_k\{Z_1, Z_2, \cdots, Z_r\} \subset \mathcal{X} = \mathcal{F}_k\{X_1, X_2, \cdots, X_n\}$$

为一 k 值逻辑子空间. $\mathcal{Z}$ 称为一个 k 值逻辑正规子空间, 如果存在 $\{Z_{r+1}, Z_{r+2}, \cdots, Z_n\} \subset \mathcal{X}$ 使得

$$T : (X_1, X_2, \cdots, X_n) \mapsto (Z_1, Z_2, \cdots, Z_n)$$

为一坐标变换.

类似于布尔网络的正规子空间, 容易证明下面的定理.

定理 6.4.3　考察 k 值逻辑网络 (6.4.1) (或 k 值逻辑控制网络 (6.4.2)). 设 $\mathcal{Z} = \mathcal{F}_k\{Z_1, Z_2, \cdots, Z_r\}$, 其结构矩阵为 $M_{\mathcal{Z}} \in \mathcal{L}_{k^r \times k^n}$. $\mathcal{Z}$ 为一个 r 维 k 值逻辑正规子空间, 且 $\{Z_1, Z_2, \cdots, Z_r\}$ 为其基底, 当且仅当,

$$\sum_{i=1}^{k^n} \mathrm{Col}_i(M_{\mathcal{Z}}) = k^{n-r} \mathbf{1}_{k^r}. \tag{6.4.9}$$

换言之, $M_{\mathcal{Z}}$ 具有 k^r 组互不相同的列: $\delta_{k^r}^i$, $i = 1, 2, \cdots, k^r$, 而每一组的个数均为 k^{n-r}.

定义 6.4.4　考察 k 值逻辑网络 (6.4.1). 子空间 $\mathcal{Z}^1 = \mathcal{F}_k\{Z^1\} = \mathcal{F}_k\{Z_1^1, Z_2^1, \cdots, Z_r^1\}$ 称为系统 (6.4.1) 的 k 值逻辑不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且, 在 Z 坐标下系统可表示为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t)), & Z^1(t) \in \mathcal{D}_k^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t)), & Z^2(t) \in \mathcal{D}_k^{n-r}. \end{cases} \tag{6.4.10}$$

设式 (6.4.1) 的代数状态空间表示为

$$x(t+1) = Mx(t). \tag{6.4.11}$$

类似于布尔网络, 可证明如下结论.

定理 6.4.4　考察 k 值逻辑网络 (6.4.1) 及其代数形式 (6.4.11). 设 $\mathcal{Z}^1 = \mathcal{F}_k\{Z_1^1, Z_2^1, \cdots, Z_r^1\}$ 为正规子空间, 其代数表达式为

$$z^1 = Qx, \tag{6.4.12}$$

这里 $z^1 = \ltimes_{i=1}^r z_i^1$, $Q \in \mathcal{L}_{k^r \times k^n}$. 那么, $\mathcal{Z}^1$ 为系统 (6.4.1) 的不变子空间, 当且仅当, 存在 $H \in \mathcal{L}_{k^r \times k^r}$ 使得

$$QM = HQ. \tag{6.4.13}$$

定义 6.4.5　给定 k 值逻辑控制网络 (6.4.2).

(i) $\mathcal{Z}^1 = \mathcal{F}_k\{Z^1\} = \mathcal{F}_\ell\{Z_1^1, Z_2^1, \cdots, Z_r^1\}$ 称为系统 (6.4.2) 的 X 不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且, 在 Z 坐标下系统可表示为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t)), & Z^1(t) \in \mathcal{D}^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t), U(t)), & Z^2(t) \in \mathcal{D}^{n-r}. \end{cases} \tag{6.4.14}$$

(ii) $\mathcal{Z}^1$ 称为系统 (6.4.2) 的 UX 不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且, 在 Z 坐标下系统可表示为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t), U(t)), & Z^1(t) \in \mathcal{D}^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t), U(t)), & Z^2(t) \in \mathcal{D}^{n-r}. \end{cases} \tag{6.4.15}$$

(iii) $\mathcal{Z}^1$ 称为系统 (6.4.2) 的控制 X 不变子空间, 如果存在 $Z^2 = \{Z_1^2, Z_2^2, \cdots, Z_{n-r}^2\}$, 使 $X \mapsto Z = \{Z^1, Z^2\}$ 为一坐标变换, 并且存在一个定常控制 $U(t) = U$, 使在这个控制下 $\mathcal{Z}^1$ 成为一不变子空间, 即系统变为

$$\begin{cases} Z^1(t+1) = \tilde{F}^1(Z^1(t)) & Z^1(t) \in \mathcal{D}^r, \\ Z^2(t+1) = \tilde{F}^2(Z(t)), & Z^2(t) \in \mathcal{D}^{n-r}. \end{cases} \tag{6.4.16}$$

设式 (6.4.2) 的代数状态空间表示为

$$x(t+1) = Mu(t)x(t). \tag{6.4.17}$$

那么, 有如下结论.

命题 6.4.1　考察 k 值网络 (6.4.2), 其代数状态空间表示为 (6.4.17). 设正规子空间 $\mathcal{Z}^1 \subset \mathcal{X}$ 的向量表达为

$$z^1 = Qx.$$

(i) $\mathcal{Z}^1$ 为 X 不变子空间, 当且仅当, 存在 $H \in \mathcal{L}_{k^r \times k^r}$ 使得

$$QM = H(\mathbf{1}_{k^m}^{\mathrm{T}} \otimes Q). \tag{6.4.18}$$

(ii) $\mathcal{Z}^1$ 为 UX 不变子空间, 当且仅当, 存在 $H \in \mathcal{L}_{k^r \times k^{m+r}}$ 使得

$$QM = HQ. \tag{6.4.19}$$

(iii) 将 M 均分为 k^m 块, 即

$$M = [M_1, M_2, \cdots, M_{k^m}],$$

这里 $M_i = M\delta_{k^m}^i \in \mathcal{L}_{k^n \times k^n}$, $i = 1, 2, \cdots, k^m$. $\mathcal{Z}^1$ 为控制 X 不变子空间, 当且仅当, 存在 $1 \leqslant i \leqslant k^m$ 使得 $\mathcal{Z}^1$ 为

$$x(t+1) = M_i x(t)$$

的不变子空间. 换言之, 存在一个 M_i 和一个 $H \in \mathcal{L}_{k^r \times k^r}$ 使得

$$QM_i = HQ.$$

下面讨论一个三值逻辑网络的例子.

例 6.4.1　考察一个三值逻辑网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), X_3(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), X_3(t)), \\ X_3(t+1) = F_3(X_1(t), X_2(t), X_3(t)), \end{cases} \tag{6.4.20}$$

这里 $X_i \in \mathcal{D}_3$, $F_i : \mathcal{D}_3^3 \to \mathcal{D}$, 其结构矩阵为 M_i^x, $i = 1, 2, 3$, 且

$$\begin{aligned} M_1^x &= \delta_3[3,1,1,1,1,1,1,2,2,2,1,1,1,2,2,2,2,2,2,1,1,1,2,2,2,3,3], \\ M_2^x &= \delta_3[3,2,2,2,3,3,3,1,1,1,2,2,2,2,2,2,3,3,3,3,3,3,3,3,3,3,3], \\ M_3^x &= \delta_3[2,1,1,1,1,1,1,1,1,1,3,3,3,3,3,3,3,3,3,2,2,2,2,2,2,2,2]. \end{aligned}$$

于是 (6.4.20) 的代数状态空间表示为

$$x(t+1) = M^x x(t), \tag{6.4.21}$$

这里

$$\begin{aligned}M^x &= M_1^x * M_2^x * M_3^x \\ &= \delta_{27}[26,4,4,4,7,7,7,10,10,10,6,6,6,15, \\ &\qquad 15,15,18,18,18,8,8,8,17,17,17,26,26].\end{aligned}$$

(i) 求 (6.4.20) 的标准型. 利用 M^x 可算出 (6.4.20) 有六个不动点, 没有极限环. 将不动点和它的吸引域列表如下: (将状态 δ_{27}^i 简记作 i)

$$\begin{aligned} 4: &\quad 2,3; \\ 7: &\quad 5,6,11,12,13; \\ 10: &\quad 8,9,20,21,22; \\ 15: &\quad 14,16; \\ 18: &\quad 17,19,23,24,25; \\ 26: &\quad 1,27. \end{aligned}$$

于是取 $z = T^{\mathrm{T}}x$, 这里

$$\begin{aligned}T = \delta_{27}[&4,2,3,7,5,6,11,12,13,10,8,9,20,21,22 \\ &15,14,16,18,17,19,23,24,25,26,1,27].\end{aligned}$$

那么, 在 z 坐标下, 系统转移矩阵变为

$$\begin{aligned}M^z &= T^{\mathrm{T}} M^x T \\ &= \delta_{27}[1,1,1,4,4,4,6,6,6,10,10,10,11,11, \\ &\qquad 11,16,16,16,19,19,19,20,20,20,25,25,25].\end{aligned} \tag{6.4.22}$$

不难看出, M^z 是一个标准型.

(ii) 设 $\mathcal{Z}^1 = \mathcal{F}_3\{Z_1, Z_2\}$, 这里

$$\begin{aligned} Z_1 &= G_1(X_1, X_2, X_3), \\ Z_2 &= G_2(X_1, X_2, X_3), \end{aligned} \tag{6.4.23}$$

这里 G_i 的结构矩阵, 记作 N_i, 如下:

$$N_1 = \delta_3[3,1,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,2,3,3,3,3,3,3,3,3],$$

$$N_2 = \delta_3[3,1,1,1,2,2,2,3,3,3,1,1,1,2,2,2,3,3,3,1,1,1,2,2,2,3,3].$$

于是, 令

$$z^1 = z_1 z_2 := Qx,$$

可知

$$\begin{aligned} Q &= N_1 * N_2 \\ &= \delta_9[9,1,1,1,2,2,2,3,3,3,4,4,4,5,5,5,6,6,6,7,7,7,8,8,8,9,9]. \end{aligned}$$

由此可知 $\mathcal{Z}^1 = \mathcal{F}_3(Z_1, Z_2)$ 是一个正规子空间. 令

$$\begin{aligned} H &= QM^xQ^{\mathrm{T}}(QQ^{\mathrm{T}})^{-1} \\ &= \delta_9[1,2,3,2,5,6,3,6,9]. \end{aligned}$$

容易验证

$$QM_x = HQ,$$

因此, $\mathcal{Z}^1$ 是一个不变子空间.

构造 Z_3 使 $\{Z_1, Z_2, Z_3\}$ 成一新坐标. 设

$$Z_3 = G_3(X_1, X_2, X_3),$$

并记 G_3 的结构矩阵为 N_3, 利用 N_1 和 N_2, 根据定理 6.1.2 不难构造出 N_3. 它不唯一, 例如, 取

$$N_3 = \delta_3[3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1,2,3,1,2].$$

于是可得

$$z = Tx = (N_1 * N_2 * N_3)x.$$

于是

$$\begin{aligned} T &= N_1 * N_2 * N_3 \\ &= \delta_{27}[27,1,2,3,4,5,6,7,8,9,10,11,12,13, \\ &\qquad 14,15,16,17,18,19,20,21,22,23,24,25,26]. \end{aligned}$$

那么, 在 z 坐标下, 系统变为

$$z(t+1) = TM^xT^{\mathrm{T}}z(t) := M^z z(t),$$

这里

$$\begin{aligned} M^z &= TM^xT^{\mathrm{T}} \\ &= \delta_{27}[3,3,3,6,6,6,9,9,9,5,5,5,14,14, \\ &\qquad 14,17,17,17,7,7,7,16,16,16,25,25,25]. \end{aligned}$$

返回到逻辑形式, 可得

$$\begin{cases} Z_1(t+1) = Z_1(t) \vee Z_2(t), \\ Z_2(t+1) = Z_1(t) \wedge Z_2(t), \\ Z_3(t+1) = \neg Z_1(t). \end{cases} \tag{6.4.24}$$

这就证实了 $\mathcal{Z}^1 = \mathcal{F}_3\{Z_1, Z_2\}$ 是一个不变子空间.

第 7 章　状态空间的解耦

从线性系统的状态空间出现开始, 解耦问题就成为控制系统理论的一个重要组成部分. 无论是对线性系统 (见文献 [120]) 或非线性系统 (见文献 [66]), 解耦都是一个热点问题. 解耦从理论上看是揭示状态空间内部结构的一种解构方法, 从应用上看它对工程系统的控制设计与排除干扰影响等起着重要作用.

对于逻辑控制系统, 状态空间的各种解耦问题同样意义重大. 本章讨论几类最重要的解耦问题: 干扰解耦和输入输出解耦[24].

7.1　布尔网络的干扰解耦

7.1.1　干扰解耦方程与输出友好子空间

考察一个实际的工程控制系统, 系统很难避免外界因素的影响. 同样, 一个实际的逻辑系统, 如生命科学中的基因调控网络等, 也难免会受到内部或外部干扰的影响. 下面给出一个带干扰的布尔控制网络模型:

$$\begin{cases} X_1(t+1) = F_1(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t),D_1(t),\cdots,D_q(t)), \\ \qquad\qquad\qquad\qquad\cdots\cdots \\ X_n(t+1) = F_n(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t),D_1(t),\cdots,D_q(t)), \end{cases}$$
$$Y_j(t) = C_j(X_1(t),\cdots,X_n(t)),\quad j=1,\cdots,p, \tag{7.1.1}$$

这里 $D_i(t)\in\mathcal{D}$, $i=1,\cdots,q$ 称为干扰. 所谓干扰解耦问题, 就是设计控制, 使得干扰不影响输出. 先看一个例子:

例 7.1.1　考虑一个带干扰的布尔控制网络

$$\begin{cases} X_1(t+1) = X_2(t)\wedge D(t), \\ X_2(t+1) = X_3(t)\vee U_1(t), \\ X_3(t+1) = X_4(t)\wedge[(X_2(t)\to D(t))\vee U_1(t)], \\ X_4(t+1) = \neg X_3(t)\vee[D(t)\wedge U_2(t)], \end{cases}$$
$$Y(t) = X_3(t)\wedge X_4(t). \tag{7.1.2}$$

如果我们选择

$$\begin{cases} U_1(t) = X_2(t), \\ U_2(t) = 0. \end{cases} \tag{7.1.3}$$

那么, 闭环系统就变为

$$\begin{cases} X_1(t+1) = X_2(t) \wedge D(t), \\ X_2(t+1) = X_3(t) \vee X_2(t), \\ X_3(t+1) = X_4(t), \\ X_4(t+1) = \neg X_3(t), \end{cases} \\ Y(t) = X_3(t) \wedge X_4(t). \tag{7.1.4}$$

显然, 干扰不会影响输出.

形如 (7.1.3) 的控制称为状态反馈控制. 要实现状态反馈控制需要实时地知道状态. 因此, 自然就会问这样一个问题: 能否用定常控制来实现干扰解耦呢? 我们尝试以下控制

$$\begin{cases} U_1(t) = 1, \\ U_2(t) = 0. \end{cases} \tag{7.1.5}$$

那么, 控制系统就变为

$$\begin{cases} X_1(t+1) = X_2(t) \wedge D(t), \\ X_2(t+1) = 1, \\ X_3(t+1) = 1, \\ X_4(t+1) = \neg X_3(t), \end{cases} \\ Y(t) = X_3(t) \wedge X_4(t). \tag{7.1.6}$$

对这个系统, 干扰同样不会影响输出.

受上述例子的影响, 我们给出以下的定义.

定义 7.1.1 考察带干扰的布尔控制网络 (7.1.1).

(i) (7.1.1) 称为状态反馈可干扰解耦的, 如果存在状态反馈控制

$$U_\alpha(t) = \phi_\alpha(X_1(t), \cdots, X_n(t)), \quad \alpha = 1, 2, \cdots, m \tag{7.1.7}$$

和一个坐标变换 $Z=T(X)$, 使得在 Z 坐标下闭环系统变为

$$\begin{cases} Z^1(t+1)=F^1(Z(t),D(t)), \\ Z^2(t+1)=F^2(Z^2(t)), \end{cases}$$
$$Y_j(t)=C_j(Z^2(t)), \quad j=1,2,\cdots,p, \tag{7.1.8}$$

称 (7.1.8) 为干扰解耦形式.

(ii) (7.1.1) 称为定常控制可干扰解耦的, 如果存在定常控制

$$U_\alpha(t)=c_\alpha, \quad \alpha=1,2,\cdots,m, \tag{7.1.9}$$

$c_\alpha \in \mathcal{D}$ 和一个坐标变换 $Z=T(X)$, 使得在 Z 坐标下闭环系统变为干扰解耦形式.

从定义不难看出, 要解决干扰解耦问题, 需要做两件事:

(i) 找到一个正规子空间 $\mathcal{Z}^2=\mathcal{F}\{Z_1^2,Z_2^2,\cdots,Z_r^2\}$ 使得 $C_j\in\mathcal{Z}^2$, $j=1,2,\cdots,p$.

(ii) 设计控制, 使 Z^2 的方程里不含 Z^1 和干扰 D_i, $i=1,2,\cdots,q$.

定义 7.1.2 记一个布尔控制网络的状态空间为 $\mathcal{X}=\mathcal{F}_\ell\{X_1,\cdots,X_n\}$, 输出为 $Y=\{Y_1,\cdots,Y_p\}\subset\mathcal{X}$. 一个正规子空间 $\mathcal{Z}\subset\mathcal{X}$ 称为一个输出友好子空间 (或 Y 友好子空间), 如果 $Y_i\in\mathcal{Z}$, $i=1,\cdots,p$. 一个输出友好子空间称为极小输出友好子空间, 如果不存在比它维数更小的输出友好子空间.

下面讨论如何构造输出友好子空间.

首先考虑单个输出的情况. 设 Y_1 为单输出, 因为 $Y_1\in\mathcal{X}$, 记其代数状态空间表示为

$$y_1=\delta_2[i_1,i_2,\cdots,i_{2^n}]x:=Hx. \tag{7.1.10}$$

令

$$n_j=|\{k\mid i_k=j,\ 1\leqslant k\leqslant 2^n\}|, \quad j=1,2,$$

那么, 我们有如下结论.

引理 7.1.1 设 $Y=\{Y_1\}$, Y_1 的代数状态空间表示为 (7.1.10). 那么, 存在 r 维 Y 友好子空间, 当且仅当, n_1 和 n_2 具有共同因子 2^{n-r}.

证明 (必要性) 设存在 Y 友好子空间 $\mathcal{Z}=\mathcal{F}_\ell\{Z_1,\cdots,Z_r\}$, 这里 $\{Z_1,\cdots,Z_r\}$ 为其基底. 记 $z=\ltimes_{i=1}^r z_i$, 并且

$$z=T_0x,$$

$T_0=(t_{i,j})\in\mathcal{L}_{2^r\times 2^n}$. 因为 $Y\in\mathcal{Z}$, 所以

$$y=Gz=GT_0x,$$

这里 $G\in\mathcal{L}_{2\times 2^r}$. 于是 G 可表示成

$$G=\delta_2[j_1,\cdots,j_{2^r}].$$

这里

$$H=H_1=\delta_2[i_1,i_2,\cdots,i_{2^n}]=\delta_2[j_1,\cdots,j_{2^r}]T_0.$$

定义

$$m_s=|\{k\mid j_k=s,\ 1\leqslant k\leqslant 2^r\}|,\quad s=1,2.$$

利用定理 6.1.2, 直接计算可得 H_1 的 $2^{n-r}m_1$ 列等于 δ_2^1, $2^{n-r}m_2$ 列等于 δ_2^2. 即, $n_1=2^{n-r}m_1$ 及 $n_2=2^{n-r}m_2$. 结论可得.

(充分性) 设 $y=Hx$, 其中 H 的 $n_1=2^{n-r}m_1$ 列为 δ_2^1, $n_2=2^{n-r}m_2$ 列为 δ_2^2. 如果据此可构造一个 r 维 Y 友好子空间 $\mathcal{Z}$, 并且 $Y_1\in\mathcal{Z}$, 那么即可证明.

设 $\mathcal{Z}$ 的结构矩阵为 $T_0\in\mathcal{L}_{2^r\times 2^n}$, 构造如下: 令

$$\begin{aligned}J^1&=\{k|\,\mathrm{Col}_k(H)=\delta_2^1\}\\&=\{\alpha_1,\alpha_2,\cdots,\alpha_{m_12^{n-r}}\}\\&:=\{J_1^1,J_2^1,\cdots,J_{m_1}^1\},\end{aligned}\tag{7.1.11}$$

这里

$$J_i^1=\left\{\alpha_{(i-1)2^{n-r}+j}\mid j=1,2,\cdots,2^{n-r}\right\},\quad i=1,2,\cdots,m_1.$$

类似地可定义

$$\begin{aligned}J^2&=\{k|\,\mathrm{Col}_k(H)=\delta_2^2\}\\&=\{\beta_1,\beta_2,\cdots,\beta_{m_22^{n-r}}\}\\&:=\{J_1^2,J_2^2,\cdots,J_{m_2}^2\},\end{aligned}\tag{7.1.12}$$

这里

$$J_i^2=\left\{\beta_{(i-1)2^{n-r}+j}\mid j=1,2,\cdots,2^{n-r}\right\},\quad i=1,2,\cdots,m_2.$$

利用 J^1 和 J^2 可定义 T_0 如下:

$$\mathrm{Col}_i(T_0)=\begin{cases}\delta_{2^r}^{\lambda}, & i\in J_\lambda^1,\\ \delta_{2^r}^{m_1+\mu}, & i\in J_\mu^2.\end{cases}\tag{7.1.13}$$

注意到

$$|J_\lambda^1| = |J_\mu^2| = 2^{n-r}, \quad \lambda = 1, 2, \cdots, m_1; \quad \mu = 1, 2, \cdots, m_2.$$

定理 6.1.2 保证了 $\mathcal{Z}$ 是一个正规子空间.

要证明 $Y_1 \in \mathcal{Z}$, 只要构造出 G 就可以了. 不妨设

$$G = \delta_2[\underbrace{1, \cdots, 1}_{m_1}, \underbrace{2, \cdots, 2}_{m_2}]. \tag{7.1.14}$$

直接计算即得 $GT_0 = H$, 因此有

$$y_1 = Gz = GHx. \qquad \square$$

为叙述方便, 我们将形为 2^s 的因子称为 2 型因子. 在布尔网络输出友好子空间的构造中, 我们只对 2 型因子感兴趣. 同理可定义 k 型因子, 它在 k 值逻辑网络输出友好子空间的讨论中用得到.

引理 7.1.1 的充分性证明是构造性的, 它给出输出友好子空间的构造方法, 下面用一个简单例子说明.

例 7.1.2　设 $\mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, X_3\}$,

$$Y = (X_1 \wedge X_2 \wedge X_3) \vee (X_1 \wedge \neg X_2) \vee (\neg X_1 \wedge X_2 \wedge X_3) \vee (\neg X_1 \wedge \neg X_2),$$

求一个 Y 友好子空间.

容易计算 Y 的结构矩阵 H 为

$$H = \delta_2[1, 2, 1, 1, 2, 1, 1, 1].$$

于是 $n_1 = 6$, $n_2 = 2$, $2 = 2^{n-r}$ 是 $(n_1,\ n_2)$ 的 2 型公因子, 于是 $r = 2$, $m_1 = 3$, $m_2 = 1$, 并且,

$$J^1 = \{1, 3, 4, 6, 7, 8\} := \{J_1^1, J_2^1, J_3^1\},$$
$$J^2 = \{2, 5\} := \{J_1^2\}.$$

于是

$$\mathrm{Col}_i(T_0) = \begin{cases} \delta_4^1, & i = 1, 3, \\ \delta_4^2, & i = 4, 6, \\ \delta_4^3, & i = 7, 8, \\ \delta_4^4, & i = 2, 5. \end{cases}$$

$$T_0 = \delta_4[1, 4, 1, 2, 4, 2, 3, 3].$$

$$G = \delta_2[1, 1, 1, 2].$$

为得到新坐标 Z, 令

$$z_3 = M_3 x = \delta_2[1, 1, 2, 1, 2, 2, 1, 2]x,$$

则坐标变换为

$$T = T_0 * M_3 = \delta_8[1, 7, 2, 3, 8, 4, 5, 6].$$

于是有

$$\begin{aligned} z_1 &= \delta_2[1, 2, 1, 1, 2, 1, 2, 2]x, \\ z_2 &= \delta_2[1, 2, 1, 2, 2, 2, 1, 1]x, \\ z_3 &= \delta_2[1, 1, 2, 1, 2, 2, 1, 2]x. \end{aligned}$$

而输出为

$$Y = Z_1 = Z_1 \vee (\neg Z_1 \wedge Z_2).$$

从引理 7.1.1 可得如下推论.

推论 7.1.1 设 n_1 和 n_2 的最大 2 型公因子为 2^{n-r}, 那么, 极小 Y 友好子空间的维数为 r.

下面考虑多输出的情况. 令 $Y = \{Y_1, \cdots, Y_p\} \subset \mathcal{X}$ 为 p 个布尔函数, 其代数状态空间表达式为

$$y = \delta_{2^p}[i_1, i_2, \cdots, i_{2^n}]x := Hx. \tag{7.1.15}$$

记

$$n_j = |\{k | i_k = j, 1 \leqslant k \leqslant 2^n\}|, \quad j = 1, \cdots, 2^p.$$

利用与单输出类似的讨论不难证明下面的结论.

定理 7.1.1 设 $y = \ltimes_{i=1}^p y_i$ 的结构矩阵如 (7.1.15) 所示.

(i) 存在 r 维 Y 友好子空间, 当且仅当, n_j, $j = 1, \cdots, 2^p$ 具有公因子 2^{n-r}.

(ii) 设 n_j, $j = 1, \cdots, 2^p$ 的最大 2 型公因子为 2^{n-r}, 那么, 极小 Y 友好子空间的维数为 r.

下面介绍一个构造 Y 友好子空间的方法. 将这种构造方法写成如下的算法. 该算法实际上给出了定理 7.1.1 充分性的一个构造性证明.

算法 7.1.1 第一步: 设输出 $Y = \{Y_1, Y_2, \cdots, Y_p\}$, $y = \ltimes_{i=1}^p y_i$ 为其向量表示, 计算 Y 的结构矩阵

$$y = Hx := \delta_{2^p}[i_1, i_2, \cdots, i_{2^n}]x. \tag{7.1.16}$$

第二步: 计算

$$n_i = \left|\{j \mid \mathrm{Col}_j(H) = \delta_{2^p}^i\}\right|, \quad i = 1, 2, \cdots, 2^p.$$

第三步: 找出 $n_i, i = 1, 2, \cdots, 2^p$ 的 (最大) 2 型公因子 2^{n-r}, 记 $n_i = m_i 2^{n-r}$. 定义

$$\begin{aligned} J^s &= \{j \mid \mathrm{Col}_j(H) = \delta_{2^p}^s\} \\ &= \{\alpha_1, \alpha_2, \cdots, \alpha_{n_s}\} \\ &:= \{J_1^s, J_2^s, \cdots, J_{m_s}^s\}, \quad s = 1, 2, \cdots, 2^p. \end{aligned}$$

(这里, $|J_j^s| = 2^{n-r}$, $s = 1, 2, \cdots, 2^p$, $j = 1, 2, \cdots, m_s$.)

第四步: 构造 2^r 维 Y 友好子空间 $z = T_0 x$, 通过构造其结构矩阵 T_0:

$$\mathrm{Col}_i(T_0) = \begin{cases} \delta_{2^r}^j, & i \in J_j^1, \\ \delta_{2^r}^{m_1+j}, & i \in J_j^2, \\ \qquad \cdots\cdots \\ \delta_{2^r}^{m_1+m_2+\cdots+m_{2^p-1}+j}, & i \in J_j^{2^p}. \end{cases} \tag{7.1.17}$$

同时可得 $y = Qz$, 这里,

$$Q = \delta_{2^p}\left[\underbrace{1, \cdots, 1}_{m_1 2^{n-c}}, \underbrace{2, \cdots, 2}_{m_2 2^{n-c}}, \cdots, \underbrace{2^p, \cdots, 2^p}_{m_{2^p} 2^{n-c}}\right]. \tag{7.1.18}$$

注 7.1.1　(i) 在算法 7.1.1 的第三步, 如果 2^{n-r} 是 $n_i, i = 1, 2, \cdots, 2^p$ 的最大 2 型公因子, 那么, 得到的就是极小 Y 友好子空间. 如果 2^{n-r} 不是最大 2 型公因子, 那么, 算法依然有效, 只是得到的就不是极小 Y 友好子空间.

(ii) 每个 J^s 中元素的排序可以不同, 算法依然有效. 因此, 得到的 T_0 也会不同, 这说明极小 Y 友好子空间不唯一, 这就是我们称其为“极小”而非“最小”的原因.

下面给出一个多输出的例子.

例 7.1.3　设 $\mathcal{X} = \mathcal{F}_\ell\{X_1, X_2, X_3, X_4\}$.

$$\begin{aligned} Y_1 &= F_1(x_1, X_2, X_3, X_4) = (X_1 \leftrightarrow X_3) \wedge (X_2 \bar{\vee} X_4), \\ Y_2 &= F_2(X_1, X_2, X_3, X_4) = X_1 \wedge X_3. \end{aligned} \tag{7.1.19}$$

我们依算法 7.1.1 寻找极小 Y 友好子空间. 记 $y = y_1 \ltimes y_2$, $x = \ltimes_{i=1}^4 x_i$, 容易算得 $y = Hx$, 这里

$$H = \delta_4[3, 1, 4, 4, 1, 3, 4, 4, 4, 4, 4, 2, 4, 4, 2, 4].$$

由 H 可得 $n_1 = n_2 = n_3 = 2$, $n_4 = 10$. 它们的最大 2 型因子是 $2 = 2^{n-r}$, 因此, 极小 Y 友好子空间维数为 $r = 3$. 并且 $m_1 = m_2 = m_3 = 1$, $m_4 = 5$. 为构造 T_0 计算

$$
\begin{aligned}
&J^1 = \{2, 5\} := \{J_1^1\},\\
&J^2 = \{12, 15\} := \{J_1^2\},\\
&J^3 = \{1, 6\} := \{J_1^3\},\\
&J^4 = \{3, 4, 7, 8, 9, 10, 11, 13, 14, 16\} := \{J_1^4, J_2^4, J_3^4, J_4^4, J_5^4\}.
\end{aligned}
$$

于是有

$$
\begin{aligned}
&\mathrm{Col}_2(T_0) = \mathrm{Col}_5(T_0) = \delta_8^1,\\
&\mathrm{Col}_{12}(T_0) = \mathrm{Col}_{15}(T_0) = \delta_8^2,\\
&\mathrm{Col}_1(T_0) = \mathrm{Col}_6(T_0) = \delta_8^3,\\
&\mathrm{Col}_3(T_0) = \mathrm{Col}_4(T_0) = \delta_8^4,\\
&\mathrm{Col}_7(T_0) = \mathrm{Col}_8(T_0) = \delta_8^5,\\
&\mathrm{Col}_9(T_0) = \mathrm{Col}_{10}(T_0) = \delta_8^6,\\
&\mathrm{Col}_{11}(T_0) = \mathrm{Col}_{13}(T_0) = \delta_8^7,\\
&\mathrm{Col}_{14}(T_0) = \mathrm{Col}_{16}(T_0) = \delta_8^8.
\end{aligned}
$$

可得

$$
T_0 = \delta_8[3, 1, 4, 4, 1, 3, 5, 5, 6, 6, 7, 2, 7, 8, 2, 8]. \tag{7.1.20}
$$

最后依 (7.1.18) 可知

$$
G = \delta_4[1, 2, 3, 4, 4, 4, 4, 4]. \tag{7.1.21}
$$

为检验结果, 我们将极小 Y 友好子空间 $\mathcal{Z}$ 构造出来. 首先

$$
z = T_0 x.
$$

记 $z_i := E_i x$, $i = 1, 2, 3$. 那么, E_i 可计算如下:

$$
\begin{aligned}
E_1 &= (I_2 \otimes \mathbf{1}_4^{\mathrm{T}})T_0\\
&= \delta_2[1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 1, 2, 2, 1, 2];\\
E_2 &= (\mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_2^{\mathrm{T}})T_0\\
&= \delta_2[2, 1, 2, 2, 1, 2, 1, 1, 1, 1, 2, 1, 2, 2, 1, 2];\\
E_3 &= (\mathbf{1}_4^{\mathrm{T}} \otimes I_2)T_0\\
&= \delta_2[1, 1, 2, 2, 1, 1, 1, 1, 2, 2, 1, 2, 1, 2, 2, 2].
\end{aligned} \tag{7.1.22}
$$

返回逻辑形式, 可知

$$\begin{aligned}Z_1 =& \{X_1 \wedge [X_2 \vee (\neg X_2 \wedge X_3)]\}\\ &\vee \{\neg X_1 \wedge ([X_2 \wedge \neg (X_3 \vee X_4)] \vee [\neg X_2 \wedge (\neg X_3 \wedge X_4)])\}\,;\\ Z_2 =& \{x_1 \wedge [(X_2 \wedge (X_3 \wedge \neg X_4)) \vee (\neg X_2 \wedge (X_3 \to X_4))]\}\\ &\vee \{\neg X_1 \wedge [(X_2 \wedge (X_3 \vee (\neg X_3 \wedge \neg X_4))) \vee (\neg X_2 \wedge (\neg X_3 \wedge X_4))]\}\,;\\ Z_3 =& \{X_1 \wedge [(X_2 \wedge X_3) \vee \neg X_2]\}\\ &\vee \{\neg X_1 \wedge [(X_2 \wedge (\neg X_3 \wedge X_4)) \vee (\neg X_2 \wedge (X_3 \wedge X_4))]\}\,.\end{aligned} \tag{7.1.23}$$

类似地, 可以算得

$$y_1 = \delta_2[1,1,2,2,2,2,2,2]z;$$
$$y_2 = \delta_2[1,2,1,2,2,2,2,2]z.$$

返回逻辑形式有

$$\begin{aligned}Y_1 &= Z_1 \wedge Z_2;\\ Y_2 &= Z_1 \wedge Z_3.\end{aligned} \tag{7.1.24}$$

7.1.2　状态反馈控制的设计

前面讨论了输出友好子空间的算法. 本小节讨论干扰解耦的控制设计方法. 这时我们假定一个输出友好子空间 $\mathcal{Z}^2$ 已经确定. 于是, 可以构造一个新坐标 $Z = (Z^1, Z^2)$ 使 $\mathcal{Z}^2 = \mathcal{F}_\ell\{Z^2\}$. 注意, 此时 Z 坐标并不唯一, 因为 Z^1 的选择不唯一. 但在后面的讨论中可以看到, Z^1 的选择不影响干扰解耦问题的解.

在 Z 坐标下, 系统 (7.1.1) 可表示成

$$Z^1(t+1) = F^1(Z(t), U(t), D(t)), \tag{7.1.25a}$$
$$Z^2(t+1) = F^2(Z(t), U(t), D(t)), \tag{7.1.25b}$$
$$Y(t) = h(Z^2(t)). \tag{7.1.25c}$$

将它与干扰解耦形式 (7.1.8) 比较, 显然, 干扰解耦问题就是要设计控制 $U(t)$, 使 (7.1.25b) 变为

$$F^2(Z(t), U(t), D(t)) = \tilde{F}^2(Z^2(t)). \tag{7.1.26}$$

考虑状态反馈控制, 即

$$U(t) = (U_1(X(t)), U_2(X(t)), \cdots, U_m(X(t)))\,.$$

记 $X^s=(X_1^s,\cdots,X_k^s)\subset(X_1,X_2,\cdots,X_n)$ 为 k 个基底变量. 定义 X^s 上的一族函数

$$\begin{aligned}
e_1(X^s)&=X_1^s\wedge X_2^s\wedge\cdots\wedge X_k^s,\\
e_2(X^s)&=X_1^s\wedge X_2^s\wedge\cdots\wedge\neg X_k^s,\\
e_3(X^s)&=X_1^s\wedge\cdots\wedge\neg X_{k-1}^s\wedge X_k^s,\\
e_4(X^s)&=X_1^s\wedge\cdots\wedge\neg X_{k-1}^s\wedge\neg X_k^s,\\
&\cdots\cdots\\
e_{2^k}(X^s)&=\neg X_1^s\wedge\neg X_2^s\wedge\cdots\wedge\neg X_k^s.
\end{aligned}\tag{7.1.27}$$

下面给出一个逻辑函数关于部分变量的展开式.

命题 7.1.1 设 $F(X_1,X_2,\cdots,X_n)=F(X^1,X^2)$, 这里 $X^1=(X_1,X_2,\cdots,X_k)$, $X^2=(X_{k+1},X_{k+2},\cdots,X_n)$, 那么, $F(X)$ 可以表示成关于 X^1 的展开式

$$F(X)=\bigvee_{i=1}^{2^k}e_i(X^1)\wedge Q_i(X^2).\tag{7.1.28}$$

证明 设 $F(X)$ 的结构矩阵为 $M_F=[M_1,M_2]$, 熟知

$$F(X)=(X_1\wedge F_1(X_2,\cdots,X_n))\vee(\neg X_1\wedge F_2(X_2,\cdots,X_n)),$$

这里 F_i 的结构矩阵为 M_i, $i=1,2$. 分别对 F_1 及 F_2 做关于 X_2 的剥离, 继续这个过程, 直至 X_k, 即得表达式 (7.1.28). □

注 7.1.2 记

$$M_F=[M_1,M_2,\cdots,M_{2^k}],\tag{7.1.29}$$

这里 $M_i\in\mathcal{L}_{2\times 2^{n-k}}$, $i=1,2,\cdots,2^k$. 那么, 由前面的证明可知, (7.1.28) 中的 Q_i 的结构矩阵为 M_i, $i=1,2,\cdots,2^k$.

根据命题 7.1.1, (7.1.25b) 中的 F^2 的每一个方程, 记作 F_j^2, 可表示为

$$F_j^2(Z(t),U(t),D(t))=\bigvee_{i=1}^{2^k}[e_i(z^2(t))\wedge Q_j^i(Z^1(t),U(t),D(t))],\quad j=1,\cdots,k.\tag{7.1.30}$$

因为结构矩阵与变量顺序有关, 如果把 $F_j^2(Z(t),U(t),D(t))$ 看作 $F_j^2(Z^2(t),Z^1(t),U(t),D(t))$, 其代数状态空间表示为 $M_F^jz^2(t)z^1(t)u(t)d(t)$, 并设 M_F^j 具有 (7.1.29) 形式的分割, 则 Q_j^i 的结构矩阵为 M_i.

下面看何时等式 (7.1.26) 能够成立.

命题 7.1.2

$$F^2(Z(t),U(t),D(t))=F^2(Z^2(t)),$$

当且仅当, 在表达式 (7.1.30) 中

$$Q_j^i(Z^1(t),U(t),D(t))=\text{const}., \quad j=1,\cdots,k, \quad i=1,\cdots,2^k. \tag{7.1.31}$$

证明　充分性是显然的, 下面证明必要性.

假如存在一组 i,j 使 Q_i^j 非定常. 考察它的系数函数 e_i. 如果它关于 Z_s^2 的因子就是 Z_s^2, 令 $Z_s^2=1$, 如果它关于 Z_s^2 的因子是 $\neg Z_s^2$, 令 $Z_s^2=0$, $s=1,\cdots,k$. 于是有

$$e_i(Z^2)=1, \quad e_j(z^2)=0, \quad j\neq i.$$

现在因为 Q_i^j 非定常, 当 $Q_i^j=1$ 时, 我们有 $F_j^2=1$, 而当 $Q_i^j=0$ 时, 我们有 $F_j^2=0$. 于是, 对固定的 Z^2, F_j^2 可以有不同的值, 这说明 F_j^2 不可能仅为 Z^2 的函数. □

总结前面的讨论. 可以得到如下的状态反馈可干扰解耦的基本结论.

定理 7.1.2　考察网络 (7.1.1). 干扰解耦问题可解, 当且仅当,

(i) 存在一个输出友好子空间 $\mathcal{Z}^2$ 使得在新坐标 $Z=(Z^1,Z^2)$ 下网络 (7.1.1) 具有 (7.1.25) 的形式.

(ii) 当 (7.1.25b) 中的 F^2 表示成关于部分变量表示式 (7.1.30) 时, 存在状态反馈控制, 使得 (7.1.31) 成立.

从定理 7.1.2 可知, 干扰解耦问题现在变为: 寻找反馈控制 $U_\alpha(t)=\phi_\alpha(X(t))$, $\alpha=1,2,\cdots,m$, 使得 (7.1.31) 成立. 注意到, 有效的反馈控制应当使所有的 Q_j^i 同时均成定常值, 我们的做法是, 对每一个 Q_j^i 找到有效控制集合, 然后再找出共同的有效控制.

给定 $Q_j^i(Z^1(t),U(t),D(t))$, 如果 $d(t)=\delta_{2^q}^s$, 则记 $D(t)=D_s(t)$. 由于干扰的不确定性, 要使 Q_j^i 成定常值, 则对所有 $D_s(t)$ 它必须都是定常的, 记

$$Q_{i,j}^s:=Q_j^i(Z^1(t),U(t),D_s(t)), \quad s=1,2,\cdots,2^q.$$

那么

$$Q_j^i(Z^1(t),U(t),D(t))\simeq 1$$

就等价于

$$\bigwedge_{s=1}^{2^q} Q_{i,j}^s(Z^1(t),U(t))=1.$$

$$Q_j^i(Z^1(t),U(t),D(t))\simeq 0$$

就等价于

$$\bigvee_{s=1}^{2^q} Q_{i,j}^s(Z^1(t),U(t))=0.$$

定义

$$\begin{aligned}P_j^i(Z^1(t),U(t)) &:= \bigwedge_{s=1}^{2^q} Q_{i,j}^s(Z^1(t),U(t)),\\ N_j^i(Z^1(t),U(t)) &:= \bigvee_{s=1}^{2^q} Q_{i,j}^s(Z^1(t),U(t)),\end{aligned} \tag{7.1.32}$$

那么, 状态反馈 $U(Z^1)$ 使 Q_j^i 成定常, 就等价于它或者使 $P_j^i=1$, 或者使 $N_j^i=0$. 于是我们有以下结论.

命题 7.1.3 考察 $Q_j^i(Z^1(t),U(t),D(t))$, 状态反馈 $U(Z^1)$ 使 Q_j^i 成定常, 当且仅当, $U(Z^1)$ 或者为 $P_j^i(Z^1(t),U(t))=1$ 的前提解, 或者为 $N_j^i(Z^1(t),U(t))=0$ 的前提解.

注 7.1.3 (i) 关于前提解的定义和算法, 可回忆第 2 章. 亦可参见文献 [108]. 因为前提解是简单可算的, 命题 7.1.3 解决了状态反馈控制的设计问题.

(ii) 如果对所有 (i,j) 找不出共同的前提解, 则干扰解耦问题在该 Y 友好子空间下无解.

(iii) 在命题 7.1.3 中, 无论 Q, P, 或 N 的上、下标都是多余的, 这里加上、下标只是为了对应定理 7.1.2 的记号, 去掉上、下标, 将 Z^1 用 Z 代替, 该命题一样成立.

下面讨论一个例子.

例 7.1.4 考察如下带干扰的布尔控制网络:

$$\begin{cases}X_1(t+1)=X_4(t)\bar{\vee}U_1(t),\\ X_2(t+1)=(X_2(t)\bar{\vee}X_3(t))\wedge\neg D(t),\\ X_3(t+1)=[(X_2(t)\leftrightarrow X_3(t))\vee D(t)]\bar{\vee}[(X_1\leftrightarrow X_5)\vee U_2(t)],\\ X_4(t+1)=[U_1(t)\to(\neg X_2(t)\vee D(t))]\wedge(X_2(t)\leftrightarrow X_3(t)),\\ X_5(t+1)=(X_4(t)\bar{\vee}U_1(t))\leftrightarrow[(U_2(t)\wedge\neg X_2(t))\vee X_4(t)],\end{cases} \tag{7.1.33}$$
$$Y(t)=X_4(t)\wedge(X_1(t)\leftrightarrow X_5(t)),$$

这里, $U_1(t),U_2(t)$ 是控制, $D(t)$ 是干扰, $Y(t)$ 是输出.

记 $x(t)=\ltimes_{i=1}^5 x_i(t)$, $u=u_1(t)\ltimes u_2(t)$, 分别为状态与控制的向量表示, 则可得 (7.1.33) 的代数状态空间表达式为

$$\begin{cases}x(t+1)=Lu(t)d(t)x(t),\\ y(t)=Hx(t),\end{cases} \tag{7.1.34}$$

这里,

$$
\begin{aligned}
L = \delta_{32}[&30,30,14,14,32,32,16,16,32,32,15,15,30,30,13,13,\\
&30,30,14,14,32,32,16,16,32,32,15,15,30,30,13,13,\\
&32,32,16,16,20,20,4,4,20,20,3,3,30,30,13,13,\\
&32,32,16,16,20,20,4,4,20,20,3,3,30,30,13,13,\\
&30,26,14,10,32,28,16,12,32,28,16,12,30,26,14,10,\\
&26,30,10,14,28,32,12,16,28,32,12,16,26,30,10,14,\\
&32,28,16,12,20,24,4,8,20,24,4,8,30,26,14,10,\\
&28,32,12,16,24,20,8,4,24,20,8,4,26,30,10,14,\\
&13,13,29,29,15,15,31,31,15,15,32,32,13,13,30,30,\\
&13,13,29,29,15,15,31,31,15,15,32,32,13,13,30,30,\\
&13,13,29,29,3,3,19,19,3,3,20,20,13,13,30,30,\\
&13,13,29,29,3,3,19,19,3,3,20,20,13,13,30,30,\\
&13,9,29,25,15,11,31,27,15,11,31,27,13,9,29,25,\\
&9,13,25,29,11,15,27,31,11,15,27,31,9,13,25,29,\\
&13,9,29,25,3,7,19,23,3,7,19,23,13,9,29,25,\\
&9,13,25,29,7,3,23,19,7,3,23,19,9,13,25,29];
\end{aligned}
$$

$$
H = \delta_2[1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,1,2,2].
$$

首先, 我们找一个极小 Y 友好子空间. 观察 H 可知, $n_1 = 8$, $n_2 = 24$. 因此, 它们的最大 2 型公因子为 $2^s = 2^3$, 且 $m_1 = 1$, $m_2 = 3$. 于是, 极小 Y 友好子空间的维数为 $r = n - s = 5 - 3 = 2$. 利用算法 7.1.1 不难算得

$$
T_0 = \delta_4[1,2,3,4,1,2,3,4,1,2,3,4,1,2,3,4,2,1,4,3,2,1,4,3,2,1,4,3,2,1,4,3]
$$

且

$$
G = \delta_2[1,2,2,2].
$$

从 T_0 可以找出 $\mathcal{Z}^2$ 的基底 $\{Z_4, Z_5\}$. 设 $z_4 = M_4 x$ 及 $z_5 = M_5 x$. 那么,

$$
\begin{aligned}
M_4 &= (I_2 \otimes \mathbf{1}_2^{\mathrm{T}})T_0\\
&= \delta_2[1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0];\\
M_5 &= (\mathbf{1}_2^{\mathrm{T}} \otimes I_2)T_0
\end{aligned}
$$

$$= \delta_2[1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1].$$

下面构造其他坐标变量 $Z^1 = \{Z_1, Z_2, Z_3\}$. 设 $z_i = M_i x$, $i = 1, 2, 3$, 根据推论 6.1.1, 不难选择一组可行的 M_i 如下:

$$\begin{aligned}
M_1 &= \delta_2[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];\\
M_2 &= \delta_2[0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1];\\
M_3 &= \delta_2[1,1,1,1,2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,1,1,1,1].
\end{aligned}$$

返回逻辑形式可得

$$\begin{cases}
Z_1 = X_1,\\
Z_2 = \neg X_2,\\
Z_3 = X_2 \leftrightarrow X_3,\\
Z_4 = X_4,\\
Z_5 = X_1 \leftrightarrow X_5.
\end{cases} \tag{7.1.35}$$

记 $z = Tx$, 则

$$\begin{aligned}
T &= M_1 * M_2 * M_3 * M_4 * M_5\\
&= \delta_{32}[9,10,11,12,13,14,15,16,5,6,7,8,1,2,3,4,\\
&\qquad 26,25,28,27,30,29,32,31,22,21,24,23,18,17,20,19].
\end{aligned}$$

考虑逆变换, 则 $x = T^{\mathrm{T}} z$, 这里,

$$\begin{aligned}
T^{\mathrm{T}} = \delta_{32}[&13,14,15,16,9,10,11,12,1,2,3,4,5,6,7,8,\\
&30,29,32,31,26,25,28,27,18,17,20,19,22,21,24,23].
\end{aligned}$$

返回逻辑形式, 可得

$$\begin{cases}
X_1 = Z_1,\\
X_2 = \neg Z_2,\\
X_3 = Z_2 \bar{\vee} Z_3,\\
X_4 = Z_4,\\
X_5 = Z_1 \leftrightarrow Z_5.
\end{cases}$$

在坐标 Z 之下, 方程 (7.1.33) 的代数状态空间表达式为

$$z(t+1) = Tx(t+1) = TLu(t)\xi(t)x(t) = TLu(t)\xi(t)T^{\mathrm{T}}z(t)$$

$$= TL(I_8 \otimes T^{\mathrm{T}})u(t)\xi(t)z(t) := \tilde{L}u(t)\xi(t)z(t),$$

以及

$$y(t) = Hx(t) = HT^{\mathrm{T}}z(t) := \tilde{H}z(t),$$

这里,

$$\begin{aligned}
\tilde{L} = \delta_{32}[&17,17,1,1,19,19,3,3,17,17,2,2,19,19,4,4,\\
&17,17,1,1,19,19,3,3,17,17,2,2,19,19,4,4,\\
&17,17,1,1,27,27,11,11,19,19,4,4,27,27,12,12,\\
&17,17,1,1,27,27,11,11,19,19,4,4,27,27,12,12,\\
&17,21,2,6,19,23,4,8,17,21,2,6,19,23,4,8,\\
&17,21,2,6,19,23,4,8,17,21,2,6,19,23,4,8,\\
&17,21,2,6,27,31,12,16,19,23,4,8,27,31,12,16,\\
&17,21,2,6,27,31,12,16,19,23,4,8,27,31,12,16,\\
&1,1,17,17,3,3,19,19,1,1,18,18,3,3,20,20,\\
&1,1,17,17,3,3,19,19,1,1,18,18,3,3,20,20,\\
&1,1,17,17,11,11,27,27,1,1,18,18,11,11,28,28,\\
&1,1,17,17,11,11,27,27,1,1,18,18,11,11,28,28,\\
&1,5,18,22,3,7,20,24,1,5,18,22,3,7,20,24,\\
&1,5,18,22,3,7,20,24,1,5,18,22,3,7,20,24,\\
&1,5,18,22,11,15,28,32,1,5,18,22,11,15,28,32,\\
&1,5,18,22,11,15,28,32,1,5,18,22,11,15,28,32];
\end{aligned}$$

$$\tilde{H} = \delta_2[1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2].$$

在坐标 Z 之下, 方程 (7.1.33) 变为

$$\begin{cases}
Z_1(t+1) = Z_4(t)\bar{\vee}U_1(t),\\
Z_2(t+1) = Z_3(t)\vee D(t),\\
Z_3(t+1) = Z_5(t)\vee U_2(t),\\
Z_4(t+1) = [U_1(t)\to(Z_2(t)\vee D(t))]\wedge Z_3(t),\\
Z_5(t+1) = (U_2(t)\wedge Z_2(t))\vee Z_4(t),
\end{cases}
\qquad (7.1.36)$$

$$Y(t) = Z_4(t)\wedge Z_5(t).$$

取 $\mathcal{Z}^2=\mathcal{F}_\ell\{Z_4,Z_5\}$. 于是, 需要考虑最后两个方程. 考虑 Z_4, 因为方程右边没有 Z_4, Z_5, 于是右边函数自身就是一个 Q. 于是, 问题变成是否存在 $U_1(Z_2,Z_3)$ 使

$$P:=\bigwedge_{i=0}^{1}[U_1(t)\to(Z_2(t)\vee i)]\wedge Z_3(t)=1,$$

或者

$$N:=\bigvee_{i=0}^{1}[U_1(t)\to(Z_2(t)\vee i)]\wedge Z_3(t)=0.$$

不难算得 P 的真值矩阵 T_P 如表 7.1.1 所示.

表 7.1.1　P 的真值表

u	z_2z_3			
	δ_4^1	δ_4^2	δ_4^3	δ_4^4
δ_2^1	1	0	0	0
δ_2^2	1	0	1	0

由真值矩阵可知, $P\simeq 1$ 没有前提解.

考虑 $N=0$, 显见, 它的真值表是表 7.1.1 的非矩阵. 即

$$T_N=\neg T_P=\begin{bmatrix}0&1&1&1\\0&1&0&1\end{bmatrix}.$$

显然, 它也没有前提解. 因此, 对于这个 Y 友好子空间, 状态反馈干扰解耦无解.

如果取 Y 友好子空间 $\mathcal{Z}^2=\mathcal{F}_\ell\{Z_3,Z_4,Z_5\}$. 类似的讨论可知

$$\begin{cases}U_1(t)=Z_2(t)=\neg X_2(t),\\U_2(t)=0,\end{cases}$$

可实现干扰解耦.

7.1.3　定常控制下的干扰解耦

考察网络 (7.1.1), 记其代数状态空间表达式为

$$\begin{cases}x(t+1)=Mu(t)d(t)x(t),\\y(t)=Hx(t).\end{cases}\tag{7.1.37}$$

现在因为 $U(t)$ 只能取定常控制, 干扰解耦问题变为: 存在 $U(t)=U$, 这里, U 的向量形式为 $\vec{U}=\delta_{2^m}^i$, 使得 (7.1.8) 成立. 因为 (7.1.33) 中 $D(t)\in\mathcal{D}^q$ 是任意的, 如果将 (7.1.37) 中的 M 写成

$$M=[M_1^1,M_2^1,\cdots,M_{2^q}^1,\cdots,M_1^{2^m},M_2^{2^m},\cdots,M_{2^q}^{2^m}],\tag{7.1.38}$$

这里, $M_j^i \in \mathcal{L}_{2^n \times 2^n}$, $i = 1,2,\cdots,2^m$, $j = 1,2,\cdots,2^q$. 那么, (7.1.8) 表示, 存在 $i = i^*$, 使对所有的 j, $\mathcal{Z}^2$ 都是 $M_j^{i^*}$ 的不变子空间. 于是可得如下结果.

命题 7.1.4　考察网络 (7.1.1), 设其代数状态空间表达式为 (7.1.37), 其中 M 有块分解形式 (7.1.38). 给定 Y 友好子空间 $\mathcal{Z}^2$. 网络 (7.1.1) 在给定 Y 友好子空间下, 干扰解耦问题应用定常控制可解, 当且仅当, 存在 $i = i^*$, 使对所有的 $1 \leqslant j \leqslant 2^q$, $\mathcal{Z}^2$ 都是 $M_j^{i^*}$ 的不变子空间.

应用定理 6.2.1 到上述命题, 立刻得到以下推论.

推论 7.1.2　网络 (7.1.1) 在 Y 友好子空间 $\mathcal{Z}^2$ 下, 干扰解耦问题应用定常控制可解, 当且仅当, 存在 $i = i^*$, 使

$$QM_j^{i^*} = HQ, \quad j = 1,2,\cdots,2^q \tag{7.1.39}$$

有共同的解 $H \in \mathcal{L}_{2^k \times 2^k}$. 这里 Q 是 $\mathcal{Z}^2$ 的结构矩阵, k 是 $\mathcal{Z}^2$ 的维数.

下面给出一个例子.

例 7.1.5　考察如下系统:

$$\begin{cases} x_1(t+1) = [U(t) \wedge D(t) \wedge (X_1(t) \wedge X_2(t))] \\ \qquad\qquad \vee \{U(t) \wedge \neg D(t) \wedge [X_1(t) \vee (\neg X_1(t) \wedge \neg X_2(t))]\} \\ \qquad\qquad \vee [\neg U(t) \wedge D(t) \wedge (X_1(t) \leftrightarrow X_2(t))] \\ \qquad\qquad \vee (\neg U(t) \wedge \neg D(t) \wedge X_2(t)), \\ x_2(t+1) = [U(t) \wedge D(t) \wedge \neg(X_1(t) \vee X_2(t))] \\ \qquad\qquad \vee [U(t) \wedge \neg D(t) \wedge (X_1(t) \wedge \neg X_2(t))] \\ \qquad\qquad \vee (\neg U(t) \wedge D(t) \wedge X_1(t)) \\ \qquad\qquad \vee [\neg U(t) \wedge \neg D(t) \wedge (X_2(t) \bar{\vee} X_2(t))], \end{cases}$$
$$Y(t) = X_1(t) \leftrightarrow X_2(t). \tag{7.1.40}$$

设定常控制为 $U(t) = 1$, 那么, 不难算得, 控制系统有如下代数状态空间表达式:

$$\begin{aligned} x(t+1) &= [M_1^1, M_2^1]d(t)x(t), \\ y(t) &= Qx(t), \end{aligned} \tag{7.1.41}$$

这里

$$\begin{aligned} M_1^1 &= \delta_4[2,4,4,3], \\ M_2^1 &= \delta_4[2,1,4,2], \\ Q &= \delta_2[1,2,2,1]. \end{aligned}$$

显然, $\mathcal{Y}=\mathcal{F}_{\ell}\{Y\}$ 本身就是正规子空间, 因此, 可取 $\mathcal{Z}^2$ 的基底为 $Z_2=Y=X_1\leftrightarrow X_2$. 根据推论 7.1.2, 网络 (7.1.40) 在 Y 友好子空间 $\mathcal{Z}^2=\mathcal{Y}$ 下, 干扰解耦问题应用定常控制 $U(t)=1$ 是否可解, 决定于方程

$$\begin{cases}QM_1^1=HQ,\\QM_2^1=HQ\end{cases}$$

是否有解 $H\in\mathcal{L}_{2\times 2}$. 不难检验, $H=\delta_2[2,1]$ 为其一个解. 因此, $U(t)=1$ 解决了干扰解耦问题.

为进一步验证, 不妨选 $Z_1=X_1$, 那么, 在坐标 $Z_1=X_1$, $Z_2=Y$ 下, 受控系统变为

$$\begin{cases}Z_1(t+1)=(Z_1(t)\wedge Z_2(t)\wedge D(t))\vee[(Z_1(t)\vee Z_2(t))\wedge\neg D(t)],\\Z_2(t+1)=\neg Z_2(t),\\Y(t)=Z_2(t).\end{cases}\tag{7.1.42}$$

显然, 这是个干扰解耦的系统.

如果有一个非正规的子空间 $\mathcal{Q}$ 使得 $\mathcal{Y}\subset\mathcal{Q}$, 那么, $\mathcal{Q}$ 称为非正规 Y 友好子空间. 下面这个推论说明: 推论 7.1.2 在非正规情况下仍然成立.

推论 7.1.3 设 $\mathcal{Z}^2$ 为非正规 Y 友好子空间, 推论 7.1.2 仍然成立.

证明 利用 (7.1.39) 可知

$$\begin{aligned}z^2(t+1)&=Qx(t+1)=QM^*d(t)x(t)\\&=H[M_1^{i_*},\cdots,M_{2^q}^{i_*^*}]d(t)x(t)\\&=HQx(t)=Hz^2(t).\end{aligned}\tag{7.1.43}$$

(7.1.43) 说明 Z^2 不受干扰影响, 于是 $\mathcal{Y}\subset\mathcal{Z}^2$ 当然不受干扰影响. 即 $u^{i^*}=\delta_{2^m}^{i^*}$ 解决了干扰解耦问题. □

注 7.1.4 (i) 因为 $\mathcal{Y}$ 本身就是 Y 友好子空间 (一般不是正规子空间), 因此, 可以利用推论 7.1.3 直接检验 $\mathcal{Y}$, 看它是否为不变子空间.

(ii) 关于布尔网络干扰解耦的所有结果均可平移到 k 值逻辑系统, 只要把 2 改成 k 即可. 读者可自行检验相关结果, 这里就不赘述了.

7.1.4 状态反馈干扰解耦①

考察带干扰的布尔控制网络 (7.1.1). 构造带干扰的输入输出转移矩阵, 见带干扰的输入输出转移表 (表 7.1.2).

① 本节是书完稿后补的. 2020 年 11 月中旬, 在聊城大学矩阵半张量积理论与应用研究中心的研讨班上听到山东大学教授冯俊娥介绍本节的思路, 因感其精妙, 完善后补记在此.

根据输入输出转移表可构造关于干扰解耦的真值矩阵 (见表 7.1.3).

表 7.1.2　带干扰的输入输出转移表

$d(t)$	$u(t)x(t)$				
	$\delta_{2^m}^1\delta_{2^n}^1$	$\cdots$	$\delta_{2^m}^1\delta_{2^n}^{2^n}$	$\cdots$	$\delta_{2^m}^{2^m}\delta_{2^n}^{2^n}$
$\delta_{2^q}^1$	$y_{1,1}^1$	$\cdots$	$y_{1,2^n}^1$	$\cdots$	$y_{2^m,2^n}^1$
$\delta_{2^q}^2$	$y_{1,1}^2$	$\cdots$	$y_{1,2^n}^2$	$\cdots$	$y_{2^m,2^n}^2$
$\vdots$	$\vdots$		$\vdots$	$\vdots$	
$\delta_{2^q}^{2^q}$	$y_{1,1}^{2^q}$	$\cdots$	$y_{1,2^n}^{2^q}$	$\cdots$	$y_{2^m,2^n}^{2^q}$

表 7.1.3　干扰解耦真值表

$u(t)$	$u(t)x(t)$			
	$\delta_{2^n}^1$	$\delta_{2^n}^2$	$\cdots$	$\delta_{2^n}^{2^n}$
$\delta_{2^m}^1$	$t_{1,1}$	$t_{1,2}$	$\cdots$	$t_{1,2^n}$
$\delta_{2^m}^2$	$t_{2,1}$	$t_{2,2}$	$\cdots$	$t_{2,2^n}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\delta_{2^m}^{2^m}$	$t_{2^m,1}$	$t_{2^m,2}$	$\cdots$	$t_{2^m,2^n}$

在表 7.1.3 中

$$t_{i,j}=\begin{cases}1, & \text{表 7.1.2 中的 } \delta_{2^m}^i\delta_{2^n}^j \text{ 列有相同的 } y,\\ 0, & \text{其他}.\end{cases}$$

那么, 从构造不难得到如下结论.

定理 7.1.3　*考察带干扰的布尔控制网络 (7.1.1). 干扰解耦问题可解, 当且仅当, 存在干扰解耦真值矩阵的前提解.*

证明　(充分性) 设

$$u(t)=Lx(t) \tag{7.1.44}$$

为干扰解耦真值矩阵的前提解. 则反馈控制 (7.1.44) 保证了干扰解耦真值矩阵中相应情况是真, 则干扰解耦问题可解.

(必要性) 如果不存在前提解, 则显然不存在状态反馈控制实现解耦. 但状态反馈控制具有一般性, 因此, 没有任何控制可实现解耦. □

注意: 这是干扰解耦可解性真正的充要条件.

下面给出一个例子.

例 7.1.6　(i) 考察如下系统:

$$\begin{cases} X_1(t+1) = (X_2(t) \wedge X_3(t)) \leftrightarrow (U(t) \vee D(t)), \\ X_2(t+1) = (X_1(t) \vee X_3(t)) \to U(t), \\ X_3(t+1) = (X_1(t) \bar{\vee} X_2(t)) \vee D(t), \end{cases} \tag{7.1.45}$$
$$Y_1(t) = X_1(t) \vee X_3(t)$$
$$Y_2(t) = X_2(t) \wedge X_3(t).$$

问其干扰解耦问题是否可解?

易得其代数状态空间表示为

$$\begin{cases} x(t+1) = Md(t)u(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{7.1.46}$$

这里,

$$M = \delta_8[1,5,5,5,1,5,5,5,3,7,7,7,3,5,7,5, \\ 2,6,5,5,1,5,6,6,8,4,3,3,7,1,4,2]. \tag{7.1.47}$$

$$H = \delta_4[1,2,2,2,1,4,2,4]. \tag{7.1.48}$$

容易算出, 带干扰的输入输出转移矩阵为

$$Y_X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 1 & 2 & 1 \\ 2 & 4 & 1 & 1 & 1 & 1 & 4 & 4 & 4 & 2 & 2 & 2 & 2 & 1 & 2 & 2 \end{bmatrix},$$

表中的数字 $t_{i,j}$ 用来表示 $\delta_4^{t_i^j}$.

由 Y_X 易得, 干扰解耦真值矩阵 T 为

$$T = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}.$$

T 没有前提解, 因此, 网络 (7.1.45) 的干扰解耦问题不可解.

(ii) 考虑网络 (7.1.45). 设输出只有 Y_1. 于是, 其代数状态空间表达式中 M 没变, H 变为

$$H = \delta_2[1,1,1,1,1,2,1,2]. \tag{7.1.49}$$

容易算出, 其带干扰的输入输出转移矩阵为

$$Y_X = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

表中的数字 $t_{i,j}$ 用来表示 $\delta_2^{t_i^j}$.

由 Y_X 易得, 干扰解耦真值矩阵 T 为

$$T=\begin{bmatrix}1&0&1&1&1&1&0&0\\0&1&1&1&1&1&1&1\end{bmatrix}.$$

T 有 16 个前提解, 因此, 改造后的网络 (7.1.45) 的干扰解耦问题有如下的状态反馈干扰解耦控制:

$$\begin{aligned}
&u(\delta_8^1)=\delta_2^1,\\
&u(\delta_8^2)=\delta_2^2,\\
&u(\delta_8^3)=\delta_2^1,\quad \text{或 } u(\delta_8^3)=\delta_2^2,\\
&u(\delta_8^4)=\delta_2^1,\quad \text{或 } u(\delta_8^4)=\delta_2^2,\\
&u(\delta_8^5)=\delta_2^1,\quad \text{或 } u(\delta_8^5)=\delta_2^2,\\
&u(\delta_8^6)=\delta_2^1,\quad \text{或 } u(\delta_8^6)=\delta_2^2,\\
&u(\delta_8^7)=\delta_2^2,\\
&u(\delta_8^8)=\delta_2^2.
\end{aligned}$$

7.2　Morgan 问题

Morgan 问题在现代控制理论中或许可谓最著名的解耦问题. 它最早是由 Morgan 针对线性系统提出的[102]. 问题是: 对控制和输出进行分组, 然后寻找反馈控制, 使输入输出解耦, 即一组输入只能控制相应的一组输出, 而不影响其他组输出. 这个问题的系统归纳和早期进展可见文献 [120]. Morgan 问题也被推广到非线性系统, 相关的成果也不少[53, 105]. 这个问题长盛不衰的原因之一是至今未找到充要条件, 却又曾多次被错误地宣布已解决.

对于布尔控制网络, Morgan 问题讨论不是很多. 这里的部分内容来自文献 [49]. 不同类型的解耦问题还有许多, 例如, 状态空间的块解耦[132].

7.2.1　Morgan 问题的模型形式

定义 7.2.1　考察布尔控制网络

$$\begin{cases}
X_1(t+1)=F_1(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)),\\
X_2(t+1)=F_2(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)),\\
\qquad\qquad\cdots\cdots\\
X_n(t+1)=F_n(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)),
\end{cases}\tag{7.2.1}$$

$$Y_j(t)=C_j(X_1(t),\cdots,X_n(t)),\quad j=1,2,\cdots,p,$$

这里, $(U_1(t),\cdots,U_m(t))$ 为输入, $(Y_1(t),\cdots,Y_p(t))$ 为输出, 并设 $m\geqslant p$.

输入输出解耦问题 (也称 Morgan 问题) 定义如下: 寻找一个坐标变换 $X\to Z$ 和一个状态反馈

$$u(t)=Kv(t)z(t), \tag{7.2.2}$$

这里 $K\in\mathcal{L}_{2^m\times 2^{n+m}}$, 使得 (7.2.1) 在反馈控制下变为如下输入输出解耦的形式:

$$\begin{cases} Z^0(t+1)=F^0(Z(t),V(t)),\\ Z^1(t+1)=F^1(Z^1(t),V^1(t)),\\ Z^2(t+1)=F^2(Z^2(t),V^2(t)),\\ \qquad\qquad\cdots\cdots\\ Z^s(t+1)=F^s(Z^s(t),V^s(t)),\end{cases}$$
$$\begin{aligned} &Y^1(t)=C^1(Z^1(t)),\\ &\qquad\qquad\cdots\cdots\\ &Y^s(t)=C^s(Z^s(t)),\end{aligned} \tag{7.2.3}$$

这里,

$$Z^i=\{Z_1^i,Z_2^i,\cdots,Z_{n_i}^i\},\quad i=0,1,\cdots,s, \tag{7.2.4}$$

且 $\{Z_j^i\mid i=0,1,\cdots,s;\ j=1,2,\cdots,n_i\}$ 为一坐标.

定义 7.2.2 考察布尔控制网络. 设 Z^i, $i=1,2,\cdots,s$ 如 (7.2.4) 所示, 且 $\sum_{i=1}^s n_i\leqslant n$. 如果存在 $Z^0=\{Z_1^0,Z_2^0,\cdots,Z_{n_0}^0\}$, 这里 $\sum_{i=0}^s n_i=n$, 使得 $Z=\{Z^0,Z^1,\cdots,Z^s\}$ 为一坐标, 则 $\{Z^1,Z^2,\cdots,Z^s\}$ 称为一组相容正规子空间.

记

$$Z^i=T_ix,\quad i=0,1,\cdots,s, \tag{7.2.5}$$

则显然可知如下结论.

命题 7.2.1 记 $\mathcal{Z}^*$ 的结构矩阵为

$$M^*:=T_1*T_2*\cdots*T_s, \tag{7.2.6}$$

则 $\mathcal{Z}^j=\mathcal{F}_\ell\{Z^j\}$, $j=1,2,\cdots,s$ 为相容正规子空间, 当且仅当 $\mathcal{Z}^*$ 是一个正规子空间.

为方便计, 在本章余下部分我们假定

A1: $s = p$.

不难看出: 输入输出解耦问题可解的必要条件是要有与每个输出分别相关的一族相容正规子空间.

命题 7.2.2　设输入输出解耦问题可解, 则必存在一族相容正规子空间 $\{\mathcal{Z}^j \mid j = 1, \cdots, p\}$, 使得每一个 $\mathcal{Z}^j$ 为 Y_j 友好子空间, $j = 1, \cdots, p$.

这样一族分别与 Y_j 友好的正规子空间称为相容 Y 友好子空间.

7.2.2　相容输出解耦子空间

本小节寻找网络 (7.2.1) 的相容的 Y_j 友好子空间.

先给两个关于矩阵乘积、半张量积, 以及 Khatri-Rao 积的关系的引理.

引理 7.2.1　设 $A \in \mathcal{M}_{m\times n}$, $B \in \mathcal{M}_{n\times s}$, $C \in \mathcal{M}_{p\times q}$, $D \in \mathcal{M}_{q\times s}$, 那么,

$$(AB) * (CD) = (A \otimes C)(B * D). \tag{7.2.7}$$

证明　记 $B = [b_1, b_2, \cdots, b_s]$, $D = [d_1, d_2, \cdots, d_s]$. 注意, 对两个列向量 X 和 Y, 有

$$X \ltimes Y = X \otimes Y.$$

因此,

$$\begin{aligned}
&(AB) * (CD) \\
&= (Ab_1, Ab_2, \cdots, Ab_s) * (Cd_1, Cd_2, \cdots, Cd_s) \\
&= [(Ab_1) \ltimes (Cd_1), (Ab_2) \ltimes (Cd_2), \cdots, (Ab_s) \ltimes (Cd_s)] \\
&= [(A \otimes C)(b_1 \ltimes d_1), (A \otimes C)(b_2 \ltimes d_2), \cdots, (A \otimes C)(b_s \ltimes d_s)] \\
&= (A \otimes C)\,[(b_1 \ltimes d_1), (b_2 \ltimes d_2), \cdots, (b_s \ltimes d_s)] \\
&= (A \otimes C)(B * D).
\end{aligned}$$

□

下面这个引理之前已用过多次, 这里重述并做适当推广.

引理 7.2.2　设 $X_i \in \Delta_{k_i}$, $i = 1, \cdots, n$ 为一组列向量, 则

$$\left(\mathbf{1}^{\mathrm{T}}_{k_1\times\cdots\times k_{i-1}} \otimes I_{k_i} \otimes \mathbf{1}^{\mathrm{T}}_{k_{i+1}\times\cdots\times k_n}\right) \ltimes_{j=1}^{n} X_j = X_i. \tag{7.2.8}$$

证明　直接计算可得

$$\left(\mathbf{1}^{\mathrm{T}}_{k_1\times\cdots\times k_{i-1}} \otimes I_{k_i} \otimes \mathbf{1}^{\mathrm{T}}_{k_{i+1}\times\cdots\times k_n}\right) \ltimes_{j=1}^{n} X_j$$

$$= \left(\mathbf{1}^{\mathrm{T}}_{k_1\times\cdots\times k_{i-1}} \otimes I_{k_i} \otimes \mathbf{1}^{\mathrm{T}}_{k_{i+1}\times\cdots\times k_n}\right) \times \left(\ltimes_{j=1}^{i-1} X_j \otimes X_i \otimes \ltimes_{j=i+1}^{n} X_j\right)$$
$$= \left(\mathbf{1}^{\mathrm{T}}_{k_1\times\cdots\times k_{i-1}}(\ltimes_{j=1}^{i-1} X_j)\right) \otimes (I_{k_i} X_i) \otimes \left(\mathbf{1}^{\mathrm{T}}_{k_{i+1}\times\cdots\times k_n} \ltimes_{j=i+1}^{n} X_j\right)$$
$$= X_i. \qquad \square$$

注 7.2.1 (i) 由引理 7.2.2 的证明可见, 如果 $X_i \in \Upsilon_{k_i}$ 为概率向量 (即, $\sum_{j=1}^{k_i}(X_i)_j = 1$), $i=1,\cdots,n$, 上述引理仍然成立.

(ii) 更一般的情况是, 设 $S=\{s_1,s_2,\cdots,s_k\}\subset\{1,2,\cdots,n\}$, 这里

$$1 \leqslant s_1 < s_2 < \cdots < s_k \leqslant n.$$

那么,

$$\Pi_S \cdot \ltimes_{j=1}^{n} X_j = \ltimes_{\alpha=1}^{k} X_{s_\alpha}, \tag{7.2.9}$$

这里, $X_i \in \Delta_{k_i}$ 或 $X_i \in \Upsilon_{k_i}$,

$$\Pi_S = \bigotimes_{i=1}^{n} J_i$$

且

$$J_i = \begin{cases} I_{k_i}, & i \in S, \\ \mathbf{1}_{k_i}, & i \notin S. \end{cases}$$

如果存在一个新坐标 Z, 使系统 (7.2.1) 在这个坐标下表示为

$$\begin{cases} Z^0(t+1) = F^0(Z(t),U(t)), \\ Z^1(t+1) = F^1(Z(t),U(t)), \\ Z^2(t+1) = F^2(Z(t),U(t)), \\ \qquad\cdots\cdots \\ Z^p(t+1) = F^p(Z(t),U(t)), \end{cases} \tag{7.2.10}$$
$$Y_j(t) = C_j(Z^j(t)), \quad j=1,\cdots,p,$$

则称 (7.2.10) 为网络 (7.2.1) 的输出解耦形式. 注意, 这时 $\mathcal{Z}^j$ 为 Y_j 友好子空间. 且 $\mathcal{Z}^1$, $\mathcal{Z}^2$, $\cdots$, $\mathcal{Z}^p$ 为一组相容正规子空间, 称其为相容输出解耦子空间.

下面我们讨论: 什么时候相容输出解耦子空间存在.

因为 $Y_j \in \mathcal{X}$, $j=1,2,\cdots,p$, 可以得到它们的代数状态空间表达式

$$y_j(t) = H_j x(t), \quad j=1,2,\cdots,p.$$

设 H_j 有 n_r^j 列为 $\delta_2^r, r=1,2$, 那么, n_r^j 可用下式计算:

$$n_r^j = (\delta_2^r)^{\mathrm{T}} H_j \mathbf{1}_{2^n}, \quad j=1,2,\cdots,p.$$

根据引理 7.1.1, 如果 n_1^j 与 n_2^j 的最大 2 型因子为 2^{s_j}, 那么, 最小 Y_j 友好子空间的维数为 $n-s_j$.

下面考虑什么时候一组 Y_j 友好子空间 $\mathcal{Z}^j$, $j=1,2,\cdots,p$ 是相容的.

定义 7.2.3　设 $\mathcal{X}=\mathcal{F}_\ell\{X_1,X_2,\cdots,X_n\}$ 为一 n 维状态空间,

$$\mathcal{Z}^i=\mathcal{F}_\ell\{Z_1^i,Z_2^i,\cdots,Z_{n_i}^i\}, \quad i=1,2$$

为 $\mathcal{X}$ 的两个子空间, $\mathcal{Z}^i$ 的结构矩阵为 M_i, $i=1,2$, 即

$$z^i=M_i x, \quad i=1,2, \quad x=\ltimes_{j=1}^n x_j.$$

记

$$S_i := \left\{ j \mid \mathrm{Col}_j(M_1)=\delta_{2^{n_1}}^i \right\}, \quad i=1,2,\cdots,n_1. \tag{7.2.11}$$

$$\begin{aligned} n_j^i := \left| \left\{ s \mid \mathrm{Col}_s(M_2)=\delta_{2^{n_2}}^j,\ s\in S_i \right\} \right|, \\ i=1,2,\cdots,2^{n_1};\ j=1,2,\cdots,2^{n_2}. \end{aligned} \tag{7.2.12}$$

$\mathcal{Z}^2$ 称为正交于子空间 $\mathcal{Z}^1$, 记作 $\mathcal{Z}^2 \perp \mathcal{Z}^1$, 如果

$$n_1^i=n_2^i=\cdots=n_{2^{n_2}}^i, \quad i=1,2,\cdots,2^{n_1}. \tag{7.2.13}$$

下面给出一个简单例子.

例 7.2.1　设 $\mathcal{X}=\mathcal{F}_\ell\{X_1,X_2,X_3,X_4\}$, $\mathcal{Z}^1=\mathcal{F}_\ell\{Z_1\}$, $\mathcal{Z}^2=\mathcal{F}_\ell\{Z_2\}$. $\mathcal{Z}^i$ 的结构矩阵 M_i, $i=1,2$ 如下:

$$\begin{aligned} M_1&=\delta_2[1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2], \\ M_2&=\delta_2[1,1,2,1,2,1,2,2,2,1,2,2,2,1,1,1]. \end{aligned} \tag{7.2.14}$$

容易检验

$$n_1^1=n_2^1=4, \quad n_1^2=n_2^2=4.$$

因此, $\mathcal{Z}^2 \perp \mathcal{Z}^1$ 并且 $\mathcal{Z}^1 \perp \mathcal{Z}^2$.

根据正规子空间的定义, 不难检验以下性质:

命题 7.2.3　(i) 如果 $\mathcal{Z}^2 \perp \mathcal{Z}^1$, 那么, $\mathcal{Z}^2$ 是正规子空间 (即使 $\mathcal{Z}^1$ 不是).

(ii) 如果 $\mathcal{Z}^1$ 是正规子空间, $\mathcal{Z}^2 \perp \mathcal{Z}^1$, 那么, $\mathcal{Z}^1 \perp \mathcal{Z}^2$. 因此, 对正规子空间族, 正交是一个对称关系.

(iii) 如果 $\mathcal{Z}^1$ 是正规子空间, $\mathcal{Z}^2 \perp \mathcal{Z}^1$, 那么, $\mathcal{Z}^1$, $\mathcal{Z}^2$ 为相容正规子空间.

如果 $\mathcal{Z}^1$, $\mathcal{Z}^2$ 为相容正规子空间, 那么, 记 $\mathcal{Z}^1 \cup \mathcal{Z}^2$ 为由 $\{Z^1, Z^2\}$ 生成的子空间, 它显然是一个正规子空间.

由前面的讨论不难得到以下结论.

定理 7.2.1 设 $\mathcal{X}=\mathcal{F}_\ell\{X_1, X_2, \cdots, X_n\}$ 为状态空间, $\mathcal{Z}^i \subset \mathcal{X}$, $i=1,2,\cdots,p$ 为一族正规子空间. $\{\mathcal{Z}^i,\ i=1,2,\cdots,p\}$ 为相容正规子空间, 当且仅当,

$$\mathcal{Z}^i \perp \bigcup_{j=1}^{i-1} \mathcal{Z}^j, \quad i=2,3,\cdots,p. \tag{7.2.15}$$

一个来自欧氏空间的错觉是: $\{\mathcal{Z}^i,\ i=1,2,\cdots,p\}$ 中任何两个子空间正交, 即

$$\mathcal{Z}^i \perp \mathcal{Z}^j, \quad 1 \leqslant i,j \leqslant p,\ 且 i \neq j,$$

可导致相容性. 但其实这是不够的, 这可以从以下例子看出.

例 7.2.2 回忆例 7.2.1. 如果我们构造 $\mathcal{Z}^3=\mathcal{F}_\ell\{Z_3\}$. $\mathcal{Z}^3$ 的结构矩阵 M_3 如下:

$$M_3=\delta_2[1,1,2,1,2,2,1,2,1,2,1,1,2,1,2,2], \tag{7.2.16}$$

那么, 容易验证: $\mathcal{Z}^1 \perp \mathcal{Z}^2$, $\mathcal{Z}^1 \perp \mathcal{Z}^3$, $\mathcal{Z}^2 \perp \mathcal{Z}^3$. 但 $\mathcal{Z}^1$, $\mathcal{Z}^2$, $\mathcal{Z}^3$ 这三个子空间不是相容正交子空间.

下面给出一个网络在相容输出解耦子空间下的表示的例子.

例 7.2.3 考察以下网络:

$$\begin{cases} X_1(t+1)=(X_1(t)\vee(X_2(t)\leftrightarrow X_4(t)))\wedge U_2(t),\\ X_2(t+1)=(X_2(t)\leftrightarrow X_4(t))\leftrightarrow U_1(t),\\ X_3(t+1)=[(X_1(t)\vee(X_2(t)\leftrightarrow X_4(t)))\wedge U_2(t)]\,\bar{\vee}[(X_1(t)\vee U_2(t))\leftrightarrow U_1(t)],\\ X_4(t+1)=[(X_2(t)\leftrightarrow X_4(t))\leftrightarrow U_1(t)]\leftrightarrow[(X_2(t)\leftrightarrow X_4(t))\to X_1(t)\wedge U_2(t)], \end{cases}$$
$$\begin{aligned} &Y_1(t)=X_1(t)\wedge(X_2(t)\leftrightarrow X_4(t)),\\ &Y_2(t)=X_1(t)\bar{\vee}X_3(t). \end{aligned} \tag{7.2.17}$$

不难找到一个极小 Y_1 友好子空间

$$\begin{cases} Z_1^1=X_1,\\ Z_2^1=X_2\leftrightarrow X_4 \end{cases}$$

和一个极小 Y_2 友好子空间

$$Z^2=Z_1^2=X_1\bar{\vee}X_3.$$

容易算得

$$
\begin{aligned}
M_1 &= \delta_4[1,2,1,2,2,1,2,1,3,4,3,4,4,3,4,3],\\
M_2 &= \delta_2[2,2,1,1,1,1,2,2,1,1,2,2,1,1,2,2].
\end{aligned}
$$

显然, $M_1 \perp M_2$. 于是 $\mathcal{Z}^1 \cup \mathcal{Z}^2$ 是一个正规子空间. 再选

$$
Z^0 = Z_1^0 = X_2,
$$

于是, $Z = \{Z^0, Z^1, Z^2\}$ 成为一个新坐标系. 并且, 在输出相容输出解耦 Z 坐标下, 网络 (7.2.17) 变为

$$
\begin{cases}
Z_1(t+1) = Z_3(t) \leftrightarrow U_1(t),\\
Z_2(t+1) = (Z_2(t) \vee Z_3(t)) \wedge U_2(t),\\
Z_3(t+1) = (Z_3(t) \rightarrow Z_2(t)) \wedge U_2(t),\\
Z_4(t+1) = (U_2(t) \vee Z_2(t)) \leftrightarrow U_1(t),
\end{cases}
\\
Y_1(t) = Z_2(t) \wedge Z_3(t),\\
Y_2(t) = Z_4(t),
\tag{7.2.18}
$$

这里, $Z^0(t) = Z_1(t)$, $Z^1(t) = (Z_2(t),\ Z_3(t))$, $Z^2(t) = Z_4(t)$.

7.2.3　输入输出解耦的反馈控制实现

本小节讨论如何从相容输出解耦子空间出发, 构造网络的输入输出解耦形式. 网络 (7.2.1) 的输出解耦形式 (7.2.10) 的代数状态空间表达式为

$$
\begin{cases}
z^0(t+1) = M_0 u(t) z(t),\\
z^1(t+1) = M_1 u(t) z(t),\\
\qquad\quad \cdots\cdots\\
z^p(t+1) = M_p u(t) z(t),
\end{cases}
\\
y_j(t) = G_j z^j(t), \quad j = 1, \cdots, p,
\tag{7.2.19}
$$

这里 $\mathcal{Z}^j, j = 1, \cdots, p$ 为一族相容 Y_i 友好子空间.

设状态反馈控制为

$$
u(t) = K v(t) z(t),
\tag{7.2.20}
$$

这里, $K \in \mathcal{L}_{2^m \times 2^{n+p}}$.

定义 7.2.4 给定输出解耦形式 (7.2.10), 输入输出解耦问题是: 找到形如 (7.2.20) 的状态反馈, 使该输出解耦形式在状态反馈下变为如下的形式:

$$\begin{cases} Z^0(t+1) = F^0(Z(t), V(t)), \\ Z^1(t+1) = F^1(Z^1(t), V_1(t)), \\ Z^2(t+1) = F^2(Z^2(t), V_2(t)), \\ \qquad\cdots\cdots \\ Z^p(t+1) = F^p(Z^p(t), V_p(t)), \end{cases} \\ Y_j(t) = C_j(Z^j(t)), \quad j = 1, \cdots, p. \tag{7.2.21}$$

(7.2.21) 称为 (7.2.10) 的输入输出解耦形式.

将 (7.2.20) 代入 (7.2.19) 得到

$$\begin{aligned} z^j(t+1) &= M_j K v(t) z(t) z(t) \\ &= M_j K v(t) \,\mathrm{PR}_{2^n}\, z(t) \\ &= M_j K \left(I_{2^p} \otimes \mathrm{PR}_{2^n}\right) v(t) z(t) \\ &:= P_j v_j(t) z^j(t) \\ &= P_j \Xi_j \left(I_{2^p} \otimes \Theta_j\right) v(t) z(t), \quad j = 1, 2, \cdots, p, \end{aligned} \tag{7.2.22}$$

这里, $P_j \in \mathcal{L}_{2^{n_j} \times 2^{n_j+1}}$ 可任选, 而

$$\begin{aligned} \Xi_j &= \mathbf{1}^{\mathrm{T}}_{2^{j-1}} \otimes I_2 \otimes \mathbf{1}^{\mathrm{T}}_{2^{p-j}}, \\ \Theta_j &= \mathbf{1}^{\mathrm{T}}_{2^{n_0+n_1+\cdots+n_{j-1}}} \otimes I_{2^{n_j}} \otimes \mathbf{1}^{\mathrm{T}}_{2^{n_{j+1}+\cdots+n_p}}, \quad j = 1, \cdots, p. \end{aligned} \tag{7.2.23}$$

总结以上讨论可得如下定理.

定理 7.2.2 输出解耦形式 (7.2.10) 可转化为输入输出解耦形式 (7.2.21), 当且仅当, 存在 $K \in \mathcal{L}_{2^m \times 2^{n+m}}$, $P_j \in \mathcal{L}_{2^{n_j} \times 2^{n_j+1}}$, $j = 1, \cdots, p$, 使得

$$M_j K \left(I_{2^p} \otimes \mathrm{PR}_{2^n}\right) = P_j \Xi_j \left(I_{2^p} \otimes \Theta_j\right), \quad j = 1, \cdots, p. \tag{7.2.24}$$

这里 Ξ_j 及 Θ_j 由 (7.2.23) 定义.

例 7.2.4 回忆例 7.2.3. 根据定理 7.2.2, 网络 (7.2.18) 是输入输出可解耦的, 当且仅当, 存在 $K \in \mathcal{L}_{2^2 \times 2^6}$, $P_1 \in \mathcal{L}_{2^2 \times 2^3}$, $P_2 \in \mathcal{L}_{2^1 \times 2^2}$ 使得

$$\begin{aligned} M_1 K (I_4 \otimes \mathrm{PR}_{2^4}) &= P_1 (I_2 \otimes \mathbf{1}^{\mathrm{T}}_2)(I_4 \otimes \mathbf{1}^{\mathrm{T}}_2 \otimes I_4 \otimes \mathbf{1}^{\mathrm{T}}_2), \\ M_2 K (I_4 \otimes \mathrm{PR}_{2^4}) &= P_2 (\mathbf{1}^{\mathrm{T}}_2 \otimes I_2)(I_4 \otimes \mathbf{1}^{\mathrm{T}}_2 \otimes \mathbf{1}^{\mathrm{T}}_4 \otimes I_2). \end{aligned} \tag{7.2.25}$$

直接计算 (7.2.18) 的分量代数状态空间表达式, 进而可算出

$$\begin{aligned}M_1=\delta_4[&1,1,1,1,2,2,3,3,1,1,1,1,2,2,3,3,\\&4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,\\&1,1,1,1,2,2,3,3,1,1,1,1,2,2,3,3,\\&4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4];\\M_2=\delta_2[&1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,\\&1,1,1,1,2,2,2,2,1,1,1,1,2,2,2,2,\\&2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,\\&2,2,2,2,1,1,1,1,2,2,2,2,1,1,1,1].\end{aligned}$$

不妨选 P_1, P_2 如下:

$$P_1=\delta_4[1,1,2,3,4,4,4,4],$$

以及

$$P_2=\delta_2[1,2,2,1].$$

于是可得 (7.2.25) 的解 K 如下:

$$\begin{aligned}K=\delta_4[&1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,\\&3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,\\&2,4,2,4,4,2,4,2,2,4,2,4,4,2,4,2,\\&4,2,4,2,2,4,2,4,4,2,4,2,2,4,2,4].\end{aligned}$$

利用引理 7.2.2, 解耦控制 $U_i(t)$, $i=1,\ 2$ 可计算如下:

$$\begin{cases}u_1(t)=K_1v(t)z(t),\\u_2(t)=K_2v(t)z(t),\end{cases}$$

这里 $K_1=(I_2\otimes\mathbf{1}_2^{\mathrm{T}})K$, $K_2=(\mathbf{1}_2^{\mathrm{T}}\otimes I_2)K$.

利用 K_1 和 K_2, 可构造反馈控制如下:

$$\begin{cases}u_1(t)=[(v_2(t)\leftrightarrow z_4(t))\wedge v_1(t)]\vee[((z_2(t)\leftrightarrow z_4(t))\leftrightarrow v_2(t))\wedge\neg v_1(t)],\\u_2(t)=v_1(t).\end{cases}$$

因此, (7.2.18) 的闭环控制模式变为

$$\begin{cases} Z_1(t+1) = Z_3(t) \leftrightarrow [((V_2(t) \leftrightarrow Z_4(t)) \wedge V_1(t)) \\ \qquad\qquad \vee ((Z_2(t) \leftrightarrow Z_4(t)) \leftrightarrow V_2(t)) \wedge \neg V_1(t)], \\ Z_2(t+1) = (Z_2(t) \vee Z_3(t)) \wedge V_1(t), \\ Z_3(t+1) = (Z_3(t) \rightarrow Z_2(t)) \wedge V_1(t), \\ Z_4(t+1) = Z_4(t) \leftrightarrow V_2(t), \end{cases} \tag{7.2.26}$$
$$Y_1(t) = Z_2(t) \wedge Z_3(t),$$
$$Y_2(t) = z_4(t).$$

显然, (7.2.26) 是一个输入输出解耦形式.

下面我们将 (7.2.24) 转换为代数形式, 以便求解.

将 (7.2.24) 中各方程相乘得

$$\begin{aligned} MK\left(I_{2^p} \otimes \mathrm{PR}_{2^n}\right) = &\left(P_1\Xi_1(I_{2^p} \otimes \Theta_1)\right) * \left(P_2\Xi_2(I_{2^p} \otimes \Theta_2)\right) \\ &* \cdots * \left(P_p\Xi_p(I_{2^p} \otimes \Theta_p)\right), \end{aligned} \tag{7.2.27}$$

这里, $M = M_1 * M_2 * \cdots * M_p$. 定义

$$\begin{aligned} &P := P_1 \otimes P_2 \otimes \cdots \otimes P_p, \\ &Q := \left(\Xi_1(I_{2^p} \otimes \Theta_1)\right) * \left(\Xi_2(I_{2^p} \otimes \Theta_2)\right) * \cdots * \left(\Xi_p(I_{2^p} \otimes \Theta_p)\right). \end{aligned}$$

利用引理 7.2.1 和引理 7.2.2, 易算得

$$MK\left(I_{2^p} \otimes \mathrm{PR}_{2^n}\right) = PQ. \tag{7.2.28}$$

将 K 分解为

$$K = \Phi\Psi,$$

这里 $\Phi \in \mathcal{M}_{2^p \times 2^{p-1}}$, $\Psi \in \mathcal{M}_{2^{p-1} \times 2^{n+p}}$. 那么, 我们有如下结果.

推论 7.2.1 考察输出解耦形式 (7.2.19). 如果存在 $\Phi \in \mathcal{M}_{2^p \times 2^{p-1}}$ 及 $\Psi \in \mathcal{M}_{2^{p-1} \times 2^{n+p}}$ 使得

$$\begin{cases} M\Phi = P, \\ \Psi\left(I_{2^p} \otimes \mathrm{PR}_{2^n}\right) = Q, \end{cases} \tag{7.2.29}$$

则存在形如 (7.2.20) 形式的控制, 使 (7.2.19) 变为输入输出解耦形式.

注 7.2.2 方程 (7.2.29) 是关于 Φ 和 Ψ 的线性方程, 这里 Φ 和 Ψ 为独立的未知矩阵. 因此, 要解决输入输出解耦问题, 我们只要解由 (7.2.29) 给出的两个线性方程就可以了.

7.2.4 Morgan 问题的解

回顾 Morgan 问题, 除了对输入输出解耦的要求外, 还要求输入对输出的完全能控性. 下面先讨论输出能控性问题.

定义 7.2.5 考察网络 (7.2.1).

(i) Y_d 称为是可达的, 如果对任给状态初值, 总存在 (依赖于初值的) $T > 0$ 和控制序列 $u(0), u(1), \cdots, u(T-1)$, 使得相应轨线满足 $Y_d = Hx(T)$.

(ii) 网络 (7.2.1) 称为输出能控的, 如果任意的 $Y_d \in \mathcal{D}^p$ 均可达.

记网络 (7.2.1) 的代数状态空间表达式为

$$\begin{cases} x(t+1) = Mu(t)x(t), \\ y(t) = Hx(t). \end{cases} \tag{7.2.30}$$

构造 (7.2.1) 的输出能控性矩阵如下: 令

$$L := \bigvee_{i=1}^{2^m} M\delta_{2^m}^i.$$

则 (7.2.1) 的能控性矩阵为

$$\mathcal{C} = \bigvee_{i=1}^{2^n} M^{(i)}.$$

输出能控性矩阵定义为

$$\mathcal{C}_Y = H\mathcal{C}. \tag{7.2.31}$$

根据集合能控性的结论可知如下结论.

命题 7.2.4 考察网络 (7.2.1).

(i) $Y_d = \delta_{2^m}^{\ell}$ 可达, 当且仅当,

$$\mathrm{Row}_{\ell}(\mathcal{C}_Y) = \mathbf{1}_{2^n}^{\mathrm{T}}.$$

(ii) 网络 (7.2.1) 输出能控, 当且仅当,

$$\mathcal{C}_y = \mathbf{1}_{2^m \times 2^n}.$$

现在考虑 Morgan 问题. 因为我们要求 V_j 完全控制 Y_j. 即在 V_j 作用下 Y_j 可控. 这对每个 j 均成立, 因此, 整个网络必定是输出可控的.

命题 7.2.5 考察网络 (7.2.1). 如果 Morgan 问题可解, 则原网络必定输出能控.

直观地说, 输入输出解耦加输出能控就解决了 Morgan 问题. 于是, 下面的结论是显然的.

定理 7.2.3 考察网络 (7.2.1). Morgan 问题可解, 当且仅当, 存在一个坐标变换 $Z=T(X)$ 和一个状态反馈控制 $U(t)=G(V(t),X(t))$, 使得网络 (7.2.1) 得到输入输出解耦形式 (7.2.21), 并且, (7.2.21) 中的每个子系统

$$\begin{cases} Z^j(t+1)=F_j(Z^j(t),V_j(t)), \\ Y_j(t)=C_j(Z^j(t)), \end{cases} \tag{7.2.32}$$

$j=1,\cdots,m$ 是输出可控的.

根据 (7.2.19) 和 (7.2.22), 每个子系统有如下代数状态空间表达式:

$$\begin{aligned} &z^j(t+1)=P_jv_j(t)z^j(t), \\ &y_j(t)=G_jz^j(t). \end{aligned} \tag{7.2.33}$$

记 $\mathcal{C}_{Z^j}=\bigvee_{i=1}^{2^{n_j}}\left(\bigvee_{r=1}^{2}P_j\delta_2^r\right)^{(i)}$. 那么, 根据命题 7.2.4, (7.2.33) 是输出可控的, 当且仅当,

$$\mathcal{C}_{Y_j}=G_j\mathcal{C}_{Z^j}=\mathbf{1}_{2\times 2^{n_j}}. \tag{7.2.34}$$

例 7.2.5 考察网络 (7.2.26). 令 $Z^1=Z_2\ltimes Z_3$, $Z^2=Z_4$, 则可得两个子系统的代数状态空间表达式如下:

$$\begin{aligned} &z^1(t+1)=P_1v_1(t)z^1(t), \\ &z^2(t+1)=P_2v_2(t)z^2(t), \end{aligned}$$

这里

$$\begin{aligned} &P_1=\delta_4[1,1,2,3,4,4,4,4], \\ &P_2=\delta_2[1,2,2,1]. \end{aligned}$$

同时

$$\begin{aligned} &y_1(t)=G_1z^1(t), \\ &y_2(t)=G_2z^2(t), \end{aligned}$$

这里

$$\begin{aligned} &G_1=\delta_2[1,2,2,2], \\ &G_2=I_2. \end{aligned}$$

记

$$L_1 = \bigvee_{i=1}^{2} P_1 \delta_2^i,$$

$$L_2 = \bigvee_{i=1}^{2} P_2 \delta_2^i$$

于是可得两个子系统相应的输出能控矩阵

$$\mathcal{C}_{Y_1} = G_1 \left(\bigvee_{i=1}^{4} L_1^{(i)} \right) = \mathbf{1}_{2\times 4},$$

$$\mathcal{C}_{Y_2} = G_2 \left(\bigvee_{i=1}^{2} L_2^{(i)} \right) = \mathbf{1}_{2\times 2}.$$

因此, 这两个子系统都是输出完全能控的.

注意到, 我们从网络 (7.2.17) 出发, 经坐标变换得到输出解耦模式 (7.2.18), 再经状态反馈控制变为输入输出解耦模式 (7.2.26). 最后, 我们证明了 (7.2.26) 每一个子系统都是输出完全能控的. 于是, 原始网络 (7.2.17) 的 Morgan 问题就彻底解决了.

第 8 章　稳定性与镇定

稳定性是一般动力系统的一个重要特性, 对于控制系统, 则希望通过设计控制, 使原来不稳定的系统变为稳定系统, 这就称为镇定. 分析系统稳定性与设计镇定控制是控制理论的一个重要任务, 这对逻辑控制系统也一样.

逻辑系统的稳定性指的是逻辑系统最后是否会收敛到一个不动点? 早期对这个问题的研究并不以逻辑网络的稳定性为对象. 它只关心一个有限集合上的映射的收敛性, 称为离散迭代[110]. 当然, 逻辑网络的动态过程就是一个离散迭代, 所以, 从数学本质上看, 它们是一致的. 当然, 因为逻辑系统的吸引子除了不动点还有极限环, 所以, 考虑逻辑系统稳定性时除了不动点外也对极限环有兴趣. 至于逻辑控制系统, 因为有了控制, 则产生了镇定问题, 即是否存在控制使系统成为稳定系统, 以及如何设计控制的问题.

利用代数状态空间方法研究逻辑系统的稳定性大致有三种方法: ① 关联矩阵分析[25]; ② 基于结构矩阵的分析方法[22,44]; ③ Lyapunov 函数方法[92]. 本章只介绍第一种方法. 关于布尔控制网络的镇定问题, 利用矩阵半张量积, 通过结构矩阵研究的方法被广泛应用. 讨论的问题包括集合镇定[56]、时滞布尔网络镇定[101]、状态反馈镇定[92]、部分镇定[35]、鲁棒镇定[94,150] 等. 本章介绍的是基于 Ledley 前提解的状态反馈镇定方法, 它给出的算法是充分且必要的[67].

8.1　逻辑系统的稳定性

考虑一个混合值逻辑系统

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t)), \end{cases} \tag{8.1.1}$$

这里, $X_i \in \mathcal{D}_{k_i}$, $i = 1, 2, \cdots, n$. 当 $k_1 = k_2 = \cdots = k_n = k$ 时, 系统 (8.1.1) 变为一个 k 值逻辑系统, 当 $k = 2$ 时它成为一个布尔网络, 当考虑稳定性及镇定控制时, 它们在方法上没有多大区别, 因此, 本章就考虑最一般的情况.

定义 8.1.1　*考虑逻辑系统* (8.1.1).

(i) 系统称为点稳定 (point stability) 的, 如果从任意初始值出发, 在经过有限时间 $0 \leqslant T < \infty$ 后, 都会收敛到一个固定点, 即 $X(t) = X_0, t \geqslant T$. X_0 点称为稳定点.

(ii) 系统称为环稳定 (cycle stability) 的, 如果从任意初始值出发, 在经过有限时间 $0 \leqslant T < \infty$ 后, 都会收敛到一个极限环 C_0, 即 $X(t) \in C_0, t \geqslant T$. C_0 称为稳定环.

定义 8.1.2　设 $X = (x_{ij}), Y = (y_{ij}) \in \mathcal{B}_{m\times n}$.

(i) 称 $X \leqslant Y$ 如果 $x_{ij} \leqslant y_{ij}, \forall i, j$.

(ii) X 与 Y 的向量距离, 记作 $D_v(X,Y)$, 定义为

$$D_v(X,Y) = X \bar{\vee} Y. \tag{8.1.2}$$

根据布尔矩阵的性质以及并与交的保序性, 下面这两个命题是显然的. 它们可视作文献 [110] 中关于向量情况的推广.

命题 8.1.1　设 $A \geqslant B$ 及 $C \geqslant E$, 那么,

$$A \ltimes_{\mathcal{B}} C \geqslant B \ltimes_{\mathcal{B}} E. \tag{8.1.3}$$

命题 8.1.2　设 $X, Y, Z \in \mathcal{B}_{m\times n}$, 那么,

$$\begin{cases} D_v(X,Y) = 0 \leftrightarrow X = Y, \\ D_v(X,Y) = D_v(Y,X), \\ D_v(X,Z) \leqslant D_v(X,Y) +_{\mathcal{B}} D_v(Y,Z). \end{cases} \tag{8.1.4}$$

命题 8.1.3　设 $A,\ B \in \mathcal{B}_{m\times n}$, $C \in \mathcal{B}_{n\times p}$, 以及 $E \in \mathcal{B}_{q\times m}$, 那么

(i)
$$D_v(A \times_{\mathcal{B}} C, B \times_{\mathcal{B}} C) \leqslant D_v(A,B) \times_{\mathcal{B}} C; \tag{8.1.5}$$

(ii)
$$D_v(E \times_{\mathcal{B}} A, E \times_{\mathcal{B}} B) \leqslant E \times_{\mathcal{B}} D_v(A,B). \tag{8.1.6}$$

证明　我们只证 (8.1.5). (8.1.6) 的证明类似. 依定义, 我们只需证明, 对任何 $1 \leqslant i \leqslant n$ 及 $1 \leqslant j \leqslant n$,

$$\sum_{k=1}^{n} \mathcal{B} a_{ik} c_{kj} \ \bar{\vee} \ \sum_{k=1}^{n} \mathcal{B} b_{ik} c_{kj} \leqslant \sum_{k=1}^{n} \mathcal{B} (a_{ik} \bar{\vee} b_{ik}) c_{kj}. \tag{8.1.7}$$

显然, (8.1.7) 右式为零 ($RHS = 0$), 当且仅当,

$$a_{ik} = b_{ik} \quad \text{或} \quad c_{kj} = 0, \quad \forall k. \tag{8.1.8}$$

但不管 (8.1.8) 中的哪一种情况发生, (8.1.7) 左式均为零 ($LHS = 0$). 结论显见. □

注意, 当 C 为一向量, 即 $C = X \in \mathcal{B}_{n\times 1}$ 时, 式 (8.1.5) 变为

$$D_v(A \times_{\mathcal{B}} X, B \times_{\mathcal{B}} X) \leqslant D_v(A, B) \times_{\mathcal{B}} X. \tag{8.1.9}$$

这个等式特别有用.

下面这个例子表明, 这个不等式有时是严格不等的.

例 8.1.1 设

$$A = [1, 0, 1, 0], \quad B = [1, 1, 1, 1],$$

以及

$$X = [1, 0, 0, 1]^{\mathrm{T}}.$$

那么,

$$A \times_{\mathcal{B}} X \bar{\vee} B \times_{\mathcal{B}} X = \left([1, 0, 1, 0]\begin{bmatrix}1\\0\\0\\1\end{bmatrix}\right) \bar{\vee} \left([1, 1, 1, 1]\begin{bmatrix}1\\0\\0\\1\end{bmatrix}\right) = 1 \bar{\vee} 1 = 0;$$

$$(A \bar{\vee} B) \times_{\mathcal{B}} X = [0, 1, 0, 1]\begin{bmatrix}1\\0\\0\\1\end{bmatrix} = 1,$$

即

$$D_v(A \times_{\mathcal{B}} X, B \times_{\mathcal{B}} X) < D_v(A, B) \times_{\mathcal{B}} X.$$

一个自然的问题是: (8.1.5) 及 (8.1.6) 何时会变为等式? 实际上我们有如下结果, 它在后面会用到.

命题 8.1.4 设 $A, B \in \mathcal{B}_{m\times n}$, $C \in \mathcal{L}_{n\times p}$, 以及 $E^{\mathrm{T}} \in \mathcal{L}_{m\times q}$, 那么,

$$D_v(A \times_{\mathcal{B}} C, B \times_{\mathcal{B}} C) = D_v(A, B) \times_{\mathcal{B}} C, \tag{8.1.10}$$

以及

$$D_v(E \times_{\mathcal{B}} A, E \times_{\mathcal{B}} B) = E \times_{\mathcal{B}} D_v(A, B). \tag{8.1.11}$$

证明 我们只证 (8.1.10). 根据定义, 只要证明对任何 $C \in \mathcal{L}_{n\times 1}$ 它成立就行了. 因此, 不妨设 $C = \delta_n^i$. 那么, (8.1.10) 变为

$$\mathrm{Col}_i(A) \bar{\vee} \mathrm{Col}_i(B) = \mathrm{Col}_i(A \bar{\vee} B),$$

这显然成立. □

8.2　全局稳定性

本节讨论全局 (点) 稳定性, 即何时一个布尔网络全局收敛到一个点. 由布尔网络的拓扑结构可知, 当一个布尔网络只有一个吸引子, 且这个吸引子是不动点时, 布尔网络全局稳定. 这个条件不仅充分, 而且必要. 本节关心的是, 当一个布尔网络的信息不完全时, 或者, 更确切地说, 如果只知道网络图时, 能否判定全局稳定性? 本节使用的主要方法是向量距离, 本节的结论是对文献 [110] 中相应结论的推广.

考察布尔网络

$$\begin{cases} X_1(t+1) = F_1(X_1,\cdots,X_n), \\ X_2(t+1) = F_2(X_1,\cdots,X_n), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1,\cdots,X_n), \quad X_i \in \mathcal{D}, \end{cases} \tag{8.2.1}$$

或者一个布尔控制网络

$$\begin{cases} X_1(t+1) = F_1(X_1,\cdots,X_n,U_1,\cdots,U_m), \\ X_2(t+1) = F_2(X_1,\cdots,X_n,U_1,\cdots,U_m), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1,\cdots,X_n,U_1,\cdots,U_m). \end{cases} \tag{8.2.2}$$

记网络的状态空间为 $\mathcal{X} = \mathcal{D}^n$. 一个点 $X \in \mathcal{X}$ 记为 $X = (X_1,\cdots,X_n)^{\mathrm{T}}$. 考察一个逻辑映射 $\Phi : \mathcal{X} \to \mathcal{X}$, 记其为

$$\begin{cases} Z_1 = \Phi_1(X_1,\cdots,X_n), \\ \qquad \cdots\cdots \\ Z_n = \Phi_n(X_1,\cdots,X_n). \end{cases} \tag{8.2.3}$$

该映射可简记为

$$Z = \Phi(X), \quad \text{这里 } X, Z \in \mathcal{X}. \tag{8.2.4}$$

该映射可来自布尔网络 (8.2.1), 即

$$X_{t+1} = \Phi(X_t). \tag{8.2.5}$$

下面的定理是基本的.

定理 8.2.1 设 $X, Y \in \mathcal{X}$, $\Phi : \mathcal{X} \to \mathcal{X}$, 那么,

$$D_v(\Phi(X), \Phi(Y)) \leqslant \mathcal{I}(\Phi) \ltimes_{\mathcal{B}} D_v(X, Y), \tag{8.2.6}$$

这里 $\mathcal{I}(\Phi)$ 是 Φ 的关联矩阵.

证明 设 $\Phi = (\phi_1, \phi_2, \cdots, \phi_n)$, 记

$$\mathcal{I}(\Phi) = (b_{ij}) \in \mathcal{B}_{n\times n}.$$

利用三角不等式可知

$$\begin{aligned} D_v(\phi_i(X), \phi_i(Y)) \leqslant\ & D_v(\phi_i(x_1, \cdots, x_n), \phi_i(y_1, x_2, \cdots, x_n)) \\ & + D_v(\phi_i(y_1, x_2, \cdots, x_n), \phi_i(y_1, y_2, x_3, \cdots, x_n)) \\ & + \cdots \\ & + D_v(\phi_i(y_1, \cdots, y_{n-1}, x_n), \phi_i(y_1, \cdots, y_n)) \\ \leqslant\ & \sum_{k=1}^{n} \mathcal{B} b_{i,k} D_v(x_k, y_k). \end{aligned}$$

结论显见. □

由上述证明可知, 对于一般情况, 即 $\Phi : \mathcal{D}^n \to \mathcal{D}^m$, 不等式 (8.2.6) 也成立.

定理 8.2.2 对于映射 $\Phi : \mathcal{X} \to \mathcal{X}$, 这里, $\mathcal{X} = \mathcal{D}^n$, 如果存在一个矩阵 $M \in \mathcal{B}_{n\times n}$ 使得

$$D_v(\Phi(X), \Phi(Y)) \leqslant M \ltimes_{\mathcal{B}} D_v(X, Y), \quad \forall X, Y \in \mathcal{X}, \tag{8.2.7}$$

那么,

$$\mathcal{I}(\Phi) \leqslant M. \tag{8.2.8}$$

证明 利用反证法. 假定存在一个 M 满足 (8.2.7), 并且有一个元素 $m_{ij} < b_{ij}$. 于是有 $m_{ij} = 0$ 及 $b_{ij} = 1$. 现在, 因为 ϕ_i 仅依赖于 x_j, 于是存在 $X = (X_1, \cdots, X_j, \cdots, X_n)$ 及 $Y = (X_1, \cdots, Y_j, \cdots, X_n)$, 使得 $\phi_i(X) \neq \phi_i(Y)$, 即

$$D_v(\phi_i(X), \phi_i(Y)) = 1.$$

但是, 根据 (8.2.7) 有

$$D_v(\phi_i(X), \phi_i(Y)) \leqslant \sum_{k\neq j} \mathcal{B} m_{ik} D_v(X_k, X_k) +_{\mathcal{B}} m_{ij} D_v(X_j, Y_j) = 0,$$

矛盾. □

定理 8.2.3　设 $E, F : \mathcal{X} \to \mathcal{X}$ 为两个逻辑映射, 那么,

$$\mathcal{I}(E \circ F) \leqslant \mathcal{I}(E) \ltimes_{\mathcal{B}} \mathcal{I}(F). \tag{8.2.9}$$

证明　对任意 $X, Y \in \mathcal{X}$

$$\begin{aligned} D_v(E \circ F(X), E \circ F(Y)) &\leqslant \mathcal{I}(E) \ltimes_{\mathcal{B}} d(F(X), F(Y)) \\ &\leqslant \mathcal{I}(E) \ltimes_{\mathcal{B}} \mathcal{I}(F) \ltimes_{\mathcal{B}} D_v(x, y). \end{aligned}$$

由定理 8.2.2 立得结论. □

下面是这个定理的一个直接推论.

推论 8.2.1　设 ξ 为 (8.2.1) 的一个不动点, 那么,

$$D_v(X(k), \xi) \leqslant [\mathcal{I}(F)]^{(k)} \ltimes_{\mathcal{B}} D_v(x(0), \xi). \tag{8.2.10}$$

特别地, 如果

$$\mathrm{Col}_\lambda\left([\mathcal{I}(F)]^{(k)}\right) = 0, \quad \lambda \in \Lambda := \{j_1, \cdots, j_s\} \subset \{1, 2, \cdots, n\},$$

并且

$$X_\alpha(0) = \xi_\alpha, \quad \forall \alpha \notin \{j_1, \cdots, j_s\},$$

那么, $X(t) = \xi$, $t \geqslant k$.

定义 8.2.1　网络 (8.2.1) 称为全局稳定的, 如果它全局收敛于一个点. 换言之, 它有一个不动点, 该点是唯一的吸引子.

例 8.2.1　考察如下系统:

$$\begin{cases} X_1(t+1) = F_1(X_2(t), X_3(t)), \\ X_2(t+1) = F_2(X_4(t)), \\ X_3(t+1) = c_0, \\ X_4(t+1) = F_4(X_3(t)), \end{cases} \tag{8.2.11}$$

这里 F_1, F_2 及 F_4 可以是任何逻辑函数, 且 c_0 为一逻辑常数. 则 F 的关联矩阵为

$$\mathcal{I}(F) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

容易检验 $[\mathcal{I}(F)]^{(4)} = 0$. 设 $\mathbf{0} = (0, 0, 0, 0)$ 为网络 (8.2.11) 的不动点, 那么, 网络 (8.2.11) 全局稳定于 $\mathbf{0}$.

从上面讨论可知如下结论.

命题 8.2.1 设 $\mathbf{0}$ 为 F 的一个不动点, 并且, 存在一个整数 $k>0$ 使得

$$[\mathcal{I}(F)]^{(k)}=0, \tag{8.2.12}$$

那么 $\mathbf{0}$ 是全局吸引点.

注意到, 如果 $X_e=(e_1,e_2,\cdots,e_n)$ 是网络 (8.2.1) 的一个不动点, 那么, 上述方法也可以用于检验 X_e 是否为一全局吸引子. 作一个坐标变换

$$Z_i=\begin{cases}X_i, & e_i=0,\\ \neg X_i, & e_i=1.\end{cases} \tag{8.2.13}$$

这样, 就可将系统 (8.2.1) 变为 Z 坐标下的系统

$$Z(t+1)=\tilde{F}(Z(t)). \tag{8.2.14}$$

如果存在一个 $k>0$ 使得 $[\mathcal{I}(\tilde{F})]^{(k)}=\mathbf{0}$, 则 X_e 是系统 (8.2.1) 的一个全局收敛点.

对于布尔矩阵, 不难证明如下结果.

命题 8.2.2 设布尔矩阵 $H\in\mathcal{B}_{n\times n}$, 则以下两点等价:

(i) 存在一个 $k>0$, 使得 $H^{(k)}=0$;

(ii) 存在一个置换矩阵 P 使得 $P^{\mathrm{T}}\ltimes_{\mathcal{B}}H\ltimes_{\mathcal{B}}P$ 成为严格上三角 (或严格下三角) 矩阵.

事实上, $H=\mathcal{I}(F)$ 是一个关联矩阵, P 意味着一个变量的重排.

下面这个例子表明, 命题 8.2.1 给出的只是一个充分条件, 但不必要.

例 8.2.2 考察如下网络:

$$\begin{cases}X_1(t+1)=X_1(t)\wedge X_2(t),\\ X_2(t+1)=X_1(t)\wedge(\neg X_2(t)).\end{cases} \tag{8.2.15}$$

不难直接检验 $\mathbf{0}$ 是它的全局吸引子. 但它的关联矩阵为

$$\mathcal{I}(F)=\begin{bmatrix}1&1\\1&1\end{bmatrix},$$

并且

$$[\mathcal{I}(F)]^{(k)}=\mathcal{I}(F)\neq 0,\quad k\geqslant 1.$$

那么, 用关联矩阵的方法能否得到充要条件? 理论上是可以的.

定理 8.2.4 布尔网络 (8.2.1) 是全局稳定的, 当且仅当, 存在 $k>0$ 使得

$$\mathcal{I}(F^k)=0. \tag{8.2.16}$$

证明　(必要性) 如果系统全局稳定, 那么, 经过有限时间 T 步后 (这里, $T = T_t$ 为过渡周期), 所有的状态都收敛到 ξ. 所以当 $k \geqslant T_t$ 后式 (8.2.16) 成立.

(充分性) 设 (8.2.16) 成立. 那么, 对任何 X 均成立 $F^k(X)$ $(k > T)$ 等于一个定常值, 不妨设

$$F^k(X) = \xi, \quad X \in \mathcal{X}.$$

那么, 对 $t \geqslant k$,

$$F^t(X) = F^k(F^{t-k}(X)) = \xi. \qquad \square$$

注 8.2.1　(i) 命题 8.2.1 及此后给出的方法在实际上是有用的, 由于关联矩阵是 $n \times n$, 它是 $O(n)$ 阶的.

(ii) 在定理 8.2.4 中 F^k 不是直接可计算的, 它只能通过 F 的代数表达式, 即其结构矩阵 M_F 来计算, 其大小为 $2^n \times 2^n$. 因此, 当 n 不是很小时很难计算.

(iii) 根据定理 8.2.3, 显然

$$\mathcal{I}(F^k) \leqslant [\mathcal{I}(F)]^{(k)}, \quad k \geqslant 1. \tag{8.2.17}$$

但通常等式不成立.

定义 8.2.2　设 $F : \mathcal{D}^n \to \mathcal{D}^m$. F 称为定常映射, 如果存在 $Z_0 \in \mathcal{D}^m$ 使得

$$F(X) = Z_0, \quad \forall X \in \mathcal{D}^n. \tag{8.2.18}$$

以下的性质直接来自定义.

命题 8.2.3　(i) F 是定常映射, 当且仅当, 其结构矩阵 M_F 满足

$$\text{Col}_i(M_F) = z_0, \quad 1 \leqslant i \leqslant 2^n, \tag{8.2.19}$$

这里, $z_0 \in \Delta_{2^m}$.

(ii) $\mathcal{I}(F) = 0$, 当且仅当, F 是定常映射.

回忆命题 8.2.1. 事实上, "**0** 是不动点" 这一点对稳定不是必要的. 因为网络状态空间具有离散拓扑, 稳定性与全局收敛是一致的. 从式 (8.2.17) 可知, 条件 (8.2.12) 保证 F^s, $s \geqslant k$ 是定常的. 即, $F^s(X) = \xi, \forall X$ 及 $s \geqslant k$. 因此, 系统收敛于 ξ. 将它写成如下推论.

推论 8.2.2　考察系统 (8.2.1). 如果条件 (8.2.12) 成立, 则系统全局稳定.

命题 8.2.1 是稳定分析及镇定设计的主要工具, 因此, 有必要进一步讨论.

首先, 值得指出的是, 关联矩阵 $\mathcal{I}(F)$ 与坐标选择有关.

例 8.2.3 考察如下系统:

$$\begin{cases} X_1(t+1) = [X_1(t) \wedge (X_2(t) \bar{\vee} X_3(x))] \vee (\neg X_1(t) \wedge X_3(t)), \\ X_2(t+1) = [X_1(t) \wedge (\neg X_2(t))] \vee (\neg X_1 \wedge X_2), \\ X_3(t+1) = [X_1(t) \wedge (\neg (X_2(t) \wedge X_3(t)))] \vee [\neg X_1(t) \wedge (X_2(t) \vee X_3(t))]. \end{cases} \tag{8.2.20}$$

将 (8.2.20) 简记为

$$X(t+1) = F(x(t)).$$

容易检验 **0** 是 (8.2.20) 的不动点. 该系统的关联矩阵为

$$\mathcal{I}(F) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}.$$

不难检验, 通过坐标变换不可能将其转换为严格上 (下) 三角形. 网络 (8.2.20) 可表示为代数表达形式

$$x(t+1) = Lx(t), \tag{8.2.21}$$

这里, $x_i(t) = \vec{X}_i$, $i = 1, 2, \cdots, n$, $x(t) = x_1(t)x_2(t)x_3(t)$,

$$L = \delta_8[8, 3, 1, 5, 1, 5, 3, 8].$$

利用坐标变换

$$\begin{cases} Z_1 = [X_1 \wedge \neg(X_3)] \vee [(\neg X_1) \wedge (X_2 \bar{\vee} X_3)], \\ Z_2 = [X_1 \wedge (X_2 \bar{\vee} X_3)] \vee [(\neg X_1) \wedge X_3], \\ Z_3 = X_2, \end{cases} \tag{8.2.22}$$

在向量表达式下

$$z := z_1 z_2 z_3 = Tx,$$

这里

$$T = \delta_8[7, 1, 6, 4, 5, 3, 2, 8].$$

这样, 在 Z 坐标下我们有

$$z(t+1) = TLT^{\mathrm{T}} z(t) := \tilde{L} z(t), \tag{8.2.23}$$

这里 $\tilde{L}$ 为

$$\tilde{L}=\delta_8[6,6,5,5,7,7,8,8].$$

在 z 坐标下可得

$$\begin{cases} z_1(t+1)=M_1z(t),\\ z_2(t+1)=M_2z(t),\\ z_3(t+1)=M_3z(t), \end{cases} \tag{8.2.24}$$

这里

$$\begin{aligned} M_1&=(I_3\otimes\mathbf{1}_9^{\mathrm{T}})\tilde{L}=\delta_2[2,2,2,2,2,2,2,2];\\ M_2&=(\mathbf{1}_3^{\mathrm{T}}\otimes I_3\otimes\mathbf{1}_3^{\mathrm{T}})\tilde{L}=\delta_2[1,1,1,1,2,2,2,2];\\ M_3&=(\mathbf{1}_9^{\mathrm{T}}\otimes I_3)\tilde{L}=\delta_2[2,2,1,1,1,1,2,2]. \end{aligned}$$

将代数式 (8.2.24) 返回逻辑形式, 得

$$\begin{cases} Z_1(t+1)=0,\\ Z_2(t+1)=Z_1(t),\\ Z_3(t+1)=Z_1(t)\bar{\vee}Z_2(t). \end{cases} \tag{8.2.25}$$

系统 (8.2.25) 是系统 (8.2.20) 在 Z 坐标下的表达式, 它的关联矩阵为

$$\mathcal{I}(\tilde{F})=\begin{bmatrix}0&0&0\\1&0&0\\1&1&0\end{bmatrix},$$

它是严下三角形的. 因为 $X_1=X_2=X_3=0$ 是 (8.2.20) 的不动点, 可知系统 (8.2.20) 全局稳定于 $\mathbf{0}$.

例 8.2.3 表明, 在某些情况下坐标变换可找到适当的关联矩阵以保证全局稳定.

现在, 一个自然的问题是: 如果一个网络是全局稳定的, 是否总存在一个坐标变换, 使在这个坐标下系统的关联矩阵变为严上 (下) 三角形的?

不幸的是, 这是做不到的. 见下面这个例子.

例 8.2.4　回忆例 8.2.2. 因为 $n=2$, 这里有 $2^2!=24$ 个坐标变换, 列举如下:

$$T_1=I_2;\quad T_2=\delta_4[1,2,4,3];\quad T_3=\delta_4[1,3,2,4];\quad\cdots;\quad T_{24}=\delta_4[4,3,2,1].$$

于是有

$$T_2:\begin{cases}Z_1=X_1,\\ Z_2=X_1\leftrightarrow X_2\end{cases}\quad\Rightarrow\quad\begin{cases}Z_1(t+1)=Z_1(t)\wedge Z_2(t),\\ Z_2(t+1)=\neg Z_1(t).\end{cases}$$

$$T_3:\begin{cases}Z_1=X_2,\\ Z_2=X_1\leftrightarrow X_2\end{cases}\quad\Rightarrow\quad\begin{cases}Z_1(t+1)=Z_1(t)\wedge Z_2(t),\\ Z_2(t+1)=(\neg Z_1(t))\wedge Z_2(t).\end{cases}$$

$$\cdots\cdots$$

$$T_{24}:\begin{cases}Z_1=\neg X_1,\\ Z_2=X_2\end{cases}\quad\Rightarrow\quad\begin{cases}Z_1(t+1)=Z_1(t)\wedge Z_2(t),\\ Z_2(t+1)=Z_2(t)\rightarrow Z_1(t).\end{cases}$$

虽然我们有 24 种不同坐标下的系统表达式, 但没有一种具严格上 (下) 三角形结构.

因此, 即使在所有可能坐标下命题 8.2.1 条件均不成立, 我们还得利用定理 8.2.4 进行检验.

8.3 布尔控制网络的镇定

定义 8.3.1 考察布尔控制网络 (8.2.2).

(i) (全局) 镇定问题是指: 找到控制序列 $U(t)=(U_1(t),U_2(t),\cdots,U_m(t))$, $t=1,2,\cdots$, 使得受控系统全局稳定.

(ii) 如果存在控制

$$\begin{cases}U_1(t)=G_1(X_1(t),X_2(t),\cdots,X_n(t)),\\ U_2(t)=G_2(X_1(t),X_2(t),\cdots,X_n(t)),\\ \qquad\cdots\cdots\\ U_m(t)=G_m(X_1(t),X_2(t),\cdots,X_n(t)),\end{cases}\tag{8.3.1}$$

使闭环系统 (全局) 稳定, 则称系统 (8.2.2) 可状态反馈镇定.

利用 8.2 节的结果, 如下的充分条件是显见的.

命题 8.3.1 (i) 布尔网络 (8.2.2) 是可镇定的, 如果存在控制序列 $U(t)$, $t=0,1,\cdots$, 使在适当坐标下受控系统的关联矩阵为严格上 (下) 三角形的.

(ii) 布尔网络 (8.2.2) 是可状态反馈镇定的, 如果存在状态反馈控制 (8.3.2), 使在适当坐标下闭环系统的关联矩阵为严格上 (下) 三角形的.

下面给出一个例子.

例 8.3.1　考察下列布尔控制网络:

$$\begin{cases} X_1(t+1) = \neg X_2(t), \\ X_2(t+1) = \neg X_4(t) \leftrightarrow ((X_4(t) \wedge (X_2(t)\bar{\vee} X_3(t))) \vee U(t)), \\ X_3(t+1) = \neg((X_4(t) \wedge (X_2(t)\bar{\vee} X_3(t))) \vee U(t)), \\ X_4(t+1) = (X_4(t) \vee (X_2(t)\bar{\vee} X_3(t))) \wedge U(t). \end{cases} \tag{8.3.2}$$

(i) 开环控制. 通过尝试法, 我们找到一个坐标变换

$$\begin{cases} Z_1 = X_4, \\ Z_2 = X_2 \bar{\vee} X_3, \\ Z_3 = \neg X_3, \\ Z_4 = \neg X_1. \end{cases} \tag{8.3.3}$$

不难得到这个坐标变换的反变换

$$\begin{cases} X_1 = \neg Z_4, \\ X_2 = Z_2 \leftrightarrow Z_3, \\ X_3 = \neg Z_3, \\ X_4 = Z_1. \end{cases} \tag{8.3.4}$$

在新坐标下, 网络方程 (8.3.2) 变为

$$\begin{cases} Z_1(t+1) = (Z_1(t) \vee Z_2(t)) \wedge U(t), \\ Z_2(t+1) = \neg Z_1(t), \\ Z_3(t+1) = (Z_1(t) \wedge Z_2(t)) \vee U(t), \\ Z_4(t+1) = Z_2(t) \leftrightarrow Z_3(t). \end{cases} \tag{8.3.5}$$

显然, 如果采用控制

$$U(t) = 0, \tag{8.3.6}$$

则受控系统变为

$$\begin{cases} Z_1(t+1) = 0, \\ Z_2(t+1) = \neg Z_1(t), \\ Z_3(t+1) = (Z_1(t) \wedge Z_2(t)), \\ Z_4(t+1) = Z_2(t) \leftrightarrow Z_3(t). \end{cases} \tag{8.3.7}$$

其关联矩阵为

$$\mathcal{I}(F)=\begin{bmatrix}0&0&0&0\\1&0&0&0\\1&1&0&0\\0&1&1&0\end{bmatrix},$$

它是严格下三角的. 于是可知, 定常控制 $U(t)=0$ 镇定了系统 (8.3.2).

(ii) 状态反馈控制. 我们直接从 (8.3.5) 出发. 尝试状态反馈控制

$$U(t)=\neg Z_1(t)\wedge\neg Z_2(t), \tag{8.3.8}$$

则系统 (8.3.5) 变为

$$\begin{cases}Z_1(t+1)=1,\\Z_2(t+1)=\neg Z_1(t),\\Z_3(t+1)=0,\\Z_4(t+1)=Z_2(t)\leftrightarrow Z_3(t).\end{cases} \tag{8.3.9}$$

其关联矩阵为

$$\mathcal{I}(F)=\begin{bmatrix}0&0&0&0\\1&0&0&0\\0&0&0&0\\0&1&1&0\end{bmatrix},$$

它是严格下三角的. 返回到原坐标系, (8.3.8) 变为

$$U(t)=\neg X_4(t)\wedge\neg(X_2(t)\bar{\vee}X_3(t)). \tag{8.3.10}$$

于是可知, 状态反馈控制 (8.3.10) 镇定了系统 (8.3.2).

向量距离的方法的不足在于: ① 它只提供充分条件, 不能适用于一切网络; ② 缺少系统的设计方法, 需要尝试. 它的优点是: 在系统信息不完全的情况下, 它可能仍然有效. 这种方法是早期的结果[22, 25].

8.4 节介绍状态反馈镇定的系统设计方法, 它主要基于近期的工作[67]. 在介绍这个工作之前, 我们希望强调下面这一点: 对于布尔网络或者一般的逻辑动态系统, 状态反馈控制与开环控制是等价的. 一个状态反馈控制当然可以用一个开环控制来代替. 令 $U(t):=G(X(t)),\ t\geqslant 0$ 就可以了. 那么, 一个开环控制能否用一个状态反馈控制代替呢? 其实这也是可以的. 因为离散时间逻辑动态系统只有有限多个状态, 于是, 对一个有效的开环控制, 它不会有环路, 于是, 在每一个状态点

它有一个唯一的有效控制. 因此, t 时刻的控制 $U(t)$ 就可以用来定义这个时刻的状态反馈控制, 令 $U(X(t)) := U(t)$ 即可. 这个解释表明状态反馈控制是一个一般性的控制.

8.4 状态反馈镇定控制的设计

考察布尔控制网络 (8.2.2), 本节关心的问题是, 如果系统是可镇定的, 如何给出一套反馈控制 (8.3.1) 的设计方法, 使闭环系统全局稳定. 这里使用的工具是 Ledley 前提解. 它最早由 Ledley [82, 83] 提出. 在 Kim 的专著 [9] 中有详细介绍. 文献 [108, 109] 将其置入矩阵半张量积框架, 并给出各种计算公式及应用. 相关内容可见第 2 章.

为了理解为什么 Ledley 前提解能用于设计镇定控制, 先考虑一个简单的例子.

例 8.4.1 考察布尔控制网络

$$\begin{cases} X_1(t+1) = U(t) \to [X_1(t) \vee X_2(t)], \\ X_2(t+1) = X_1(t) \wedge U(t). \end{cases} \tag{8.4.1}$$

问其能否反馈镇定到 $(1,0)$? 令

$$\begin{cases} F_1(X_1(t), X_2(t), U(t)) := U(t) \to [X_1(t) \vee X_2(t)] = 1, \\ F_2(X_1(t), X_2(t), U(t)) := X_1(t) \wedge U(t) = 0, \end{cases} \tag{8.4.2}$$

看从 (8.4.2) 能否确定出一个状态反馈控制函数

$$U(t) = G(X_1(t), X_2(t)). \tag{8.4.3}$$

构造 (8.4.2) 的真值矩阵, 见表 8.4.1.

表 8.4.1 (8.4.2) 的真值矩阵

U	X			
	$(1,1)$	$(1,0)$	$(0,1)$	$(0,0)$
1	0	0	1	0
0	1	1	1	1

即真值矩阵为

$$T = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}. \tag{8.4.4}$$

于是可知, (8.4.2) 有两个前提解:

$$U(t) = G_1(X_1(t), X_2(t)), \quad U(t) = G_2(X_1(t), X_2(t)), \tag{8.4.5}$$

其中 G_1 与 G_2 的结构矩阵分别为

$$\begin{aligned} M_1 &= \delta_2[2, 2, 1, 2] \leqslant T, \\ M_2 &= \delta_2[2, 2, 2, 2] \leqslant T. \end{aligned} \tag{8.4.6}$$

于是, 这两个解分别为

$$\begin{aligned} &U(t) = (\neg X_1(t)) \wedge X_2(t), \\ &U(t) = 0. \end{aligned} \tag{8.4.7}$$

不难直接验证, 这两个控制均能将系统 (8.4.1) 镇定到给定点 (1, 0). 道理很简单, 前提解保证了 (8.4.2) 成立, 因此, 上述反馈控制一步就让系统到达目的地.

上述例子表明, 让动态方程直接等于目标值的逻辑等式的前提解是镇定的一个充分条件. 虽然它不是必要的, 但从上述例子可以看出通过寻找前提解的方式构造状态反馈镇定控制的合理性. 下面讨论一个必要条件.

定义 8.4.1 考察布尔控制网络 (8.2.2), 一个点 X^d 称为控制不动点, 如果存在控制 U, 使得 (8.2.2) 满足

$$F_i(X^d, U) = X_i^d, \quad i = 1, 2, \cdots, n. \tag{8.4.8}$$

下面这个必要条件是显然的.

命题 8.4.1 布尔控制网络 (8.2.2) 能反馈镇定到一个点 X^d, 则 X^d 为控制不动点.

为了得到一般状态反馈镇定控制的设计方法, 我们需要推广前提解的概念.

定义 8.4.2 考察一族逻辑等式

$$\begin{cases} F_1(X_1, X_2, \cdots, X_n; U_1, U_2, \cdots, U_m) = C_1, \\ F_2(X_1, X_2, \cdots, X_n; U_1, U_2, \cdots, U_m) = C_2, \\ \qquad\qquad \cdots\cdots \\ F_s(X_1, X_2, \cdots, X_n; U_1, U_2, \cdots, U_m) = C_s. \end{cases} \tag{8.4.9}$$

$\Omega \subset \mathcal{D}^s$ 为一给定子集.

$$\begin{cases} U_1 = G_1(X_1, X_2, \cdots, X_n), \\ U_2 = G_2(X_1, X_2, \cdots, X_n), \\ \qquad\qquad \cdots\cdots \\ U_m = G_m(X_1, X_2, \cdots, X_n) \end{cases} \tag{8.4.10}$$

称为 (8.4.9) 关于容许子集 Ω 的前提解, 如果将 (8.4.10) 代入 (8.4.9) 左式, 则得到 (8.4.9) 的右式满足 $(C_1, C_2, \cdots, C_s) \in \Omega$.

定义 8.4.3　考察一族逻辑等式 (8.4.9) 和一族逻辑函数 (8.4.10).

(i) (8.4.10) 称为 (8.4.9) 限制于 $W \in \mathcal{D}^n$ 的受限前提解, 如果仅当 $X = (X_1, X_2, \cdots, X_n) \in W$ 时 (8.4.9) 成立, 则 (8.4.10) 也成立.

(ii) 关于容许子集 Ω 的, 并受限于 W 的前提解称为关于容许子集的受限前提解.

定义 8.4.4　(i) 设 f 为容许子集 Ω 上受限于 W 的前提解, g 是容许子集 Ω 上受限于 W' 的前提解. 称 f 优于 g, 记作 $f \succ g$ (或 $g \prec f$) 如果 $W' \subset W$ 并且

$$f|_{W'} = g|_{W'}.$$

(ii) 设 f 是一个受限于 W 的前提解. f 称为极大受限前提解, 如果不存在优于它的前提解, 即如果 g 为受限于 W' 的前提解, 且 $g \succ f$, 则有 $W' = W$ 且 $g|_W = f|_W$.

注意到如果 f 是受限于 W 的前提解, 并且, f 是一个极大前提解, 那么, W 是完全由真值矩阵 T 决定的 (严格地说, W 是由 T 的所有非零列所对应的元素组成的), 它与具体的 f 无关. 因此, 把 W 称为 (Ledley) 极大集.

将对一般前提解的求解结果直接用到受限前提解, 则可得到如下结论.

定理 8.4.1　设等式 (8.4.9) 关于容许子集 $\Omega \subset \mathcal{D}^s$ 的真值矩阵为 T_Ω, 并且受限集合为 $W \subset \mathcal{D}^n$, 那么, (8.4.10) 为容许子集 Ω 上的受限于 W 的前提解, 当且仅当,

$$M_G|_W \leqslant T_\Omega|_W. \tag{8.4.11}$$

受限于 W 的前提解简称为 W 前提解.

例 8.4.2　考察下列函数

$$\begin{aligned} F_1(X_1, X_2, U) &= (X_1 \vee U) \to X_2 = \delta_2^1, \\ F_2(X_1, X_2, U) &= X_1 \wedge U = \delta_2^2. \end{aligned} \tag{8.4.12}$$

容易算得, 其结构矩阵为

$$M_F = \delta_4[1, 4, 1, 4, 2, 4, 2, 2], \tag{8.4.13}$$

且 $c = \delta_4^2$. 则其真值矩阵为

$$T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix}. \tag{8.4.14}$$

(i) 因 (8.4.12) 的真值矩阵为 (8.4.14), 易知, 它没有前提解.

(ii) 设 $w=\vec{W}=\{\delta_4^1,\delta_4^3,\delta_4^4\}$, 则 W 为一极大集. 令 $u=\vec{U}=\delta_4[2,1,2,2]x$. 由定理 8.4.1 可知, 这个 U 是 W 前提解, 它是极大前提解.

(iii) 设 $w'=\vec{W}'=\{\delta_4^1,\delta_4^3\}$, 那么, $u'=\delta_4[2,1,2,1]x$. U' 是 W' 前提解, 它不是极大前提解.

下面讨论如何利用受限前提解设计状态反馈镇定器.

首先解释一下, 状态反馈镇定与自由控制序列镇定实际上是等价的. 首先, 如果有状态反馈镇定, 当然会有自由控制序列镇定, 只要取 $U(t)=U(X(t))$ 就是了. 那么, 每个自由序列控制能用状态反馈控制来代替吗? 这其实也是可以做到的. 因为每一个布尔控制网络 (或一般的逻辑控制网络), 只有有限多个状态点. 一个有效的控制轨线不可能有重复的点. 因此, 对每一个 $X\in\mathcal{D}^n$, 都会有唯一的控制值 $U_X\in\mathcal{D}^m$. 设 $U((X))=U_X$, 即得反馈控制. 换言之, 任何镇定器都可以表示为状态反馈控制器.

下面给出一个算法, 它提供了状态反馈镇定器的完整的设计过程.

算法 8.4.1 第 0 步 (初始步): 设初始的容许集合为 $\Omega(0)=W_0=\{(C_1,C_2,\cdots,C_n)\}$, 这里, $X^d=(C_1,C_2,\cdots,C_n)$ 是系统全局收敛的目标点. 关于这个容许集合构造真值矩阵 $T_{\Omega(0)}$, 利用它找出对应于 $\Omega(0)$ 的极大集 W_1.

检验是否 $X^d\in W_1$?

(i) 如果 $X^d\notin W_1$, 则 X^d 不是控制不动点, 状态反馈镇定无解. 算法结束.

(ii) 否则 (即 $X^d\in W_1$), 令 $k=1$, 进入循环步骤.

第 k 步 (循环步骤, $k\geqslant 1$) : 令

$$\Omega(k)=W_k\backslash\{W_{k-1},\cdots,W_0\}.$$

(i) 如果 $\Omega(k)=\varnothing$, 则状态反馈镇定无解. 算法结束.

(ii) 否则 (即 $\Omega(k)\neq\varnothing$), 构造真值矩阵 $T_{\Omega(k)}$, 并利用它找出关于 $\Omega(k)$ 的极大集 W_{k+1}. 然后检验: 如果

$$\bigcup_{i=0}^{k}\Omega_i=\varDelta_{2^n}, \tag{8.4.15}$$

则镇定问题可解. 记此时的 k 为 k_*, 然后到最后一步. 否则 (即 (8.4.15) 不成立), 对下一个 k (即 $k+1$) 回到循环步骤.

最后一步 (构造反馈镇定器): 依照以下方法构造反馈镇定器的结构矩阵 $u(t)=M_Gx(t)$: 选择 M_G, 使其满足如下不等式

$$M_G|_{\Omega(i)}\leqslant\begin{cases}T_{\omega(0)}, & i=0,\\ T_{\omega(i-1)}\big|_{\Omega(i)}, & i=1,\cdots,k^*.\end{cases} \tag{8.4.16}$$

定理 8.4.2　布尔控制网络 (8.2.2) 能全局镇定到 X^d, 当且仅当, 算法 8.4.1 可以进行到底, 并产生状态反馈控制器.

证明　从第一步出发, 可以看出, $X \in \Omega(1)$, 则存在 W_0 前提解, 作为状态反馈控制, 它保证 $F_i(X,U(X)) = C_i$, $i = 1,2,\cdots,n$. 即, 状态反馈控制可将 $X \in \Omega(1)$ 驱赶到 $\Omega(0) = \{C\}$. 类似地讨论可看出, 在第 k 步, 构造的状态反馈, 当限制在 $\Omega(k)$ 上时, 它可将 $X \in \Omega(k)$ 的点驱赶到 $\Omega(k-1)$. 最后, 如果存在 k^* 使得 (8.4.15) 成立, 那么, 所有的点都会在 k^* 步内被驱赶到 C. 因此, 只要算法成功, 所有的点都会被驱赶到 C.

下面证明, 如果算法不成功, 则状态反馈镇定是不可能的. 第一种情况, C 不是控制不动点, 显然不能镇定. 第二种情况, 设 k^* 是让 $\Omega_k \neq \varnothing$ 的最后一步, 同时

$$\bigcup_{i=0}^{k^*} \Omega_i \neq \Delta_{2^n}.$$

那么, 存在 $X \in \Delta_{2^n} \Big\backslash \left\{\bigcup_{i=0}^{k^*} \Omega_i\right\}$. 于是, 不存在状态反馈控制, 使它能到达 C.

□

注 8.4.1　实际上, 算法 8.4.1 给出的是时间最优控制, 即每一点都能在最短时间到达终点. 如果不要求时间最优, 状态反馈镇定器可能更多.

下面用一个例子来描述这个算法.

例 8.4.3　考察如下的布尔网络:

$$\begin{cases} X_1(t+1) = X_2(t) \vee U_1(t), \\ X_2(t+1) = X_4(t) \vee (U_2(t) \wedge X_1(t)), \\ X_3(t+1) = (X_1(t) \wedge X_4(t))\, \bar{\vee}\, (\neg X_3(t)), \\ X_4(t+1) = (\neg X_1(t)) \leftrightarrow U_2(t). \end{cases} \tag{8.4.17}$$

讨论它是否可由状态反馈镇定到 $X^d = (1,1,0,1)$?

构造如下等式:

$$\begin{cases} F_1(X,U) = X_2 \vee U_1 = 1, \\ F_2(X,U) = X_4 \vee (U_2 \wedge X_1) = 1, \\ F_3(X,U) = (X_1 \wedge X_4)\, \bar{\vee}\, (\neg X_3) = 0, \\ F_4(X,U) = (\neg X_1) \leftrightarrow U_2 = 1. \end{cases} \tag{8.4.18}$$

设 F_i 的结构矩阵为 M_i, $i = 1,2,3,4$. 那么 $F := F_1 \ltimes F_2 \ltimes F_3 \ltimes F_4$ 的结构矩阵为

$$M_F = M_1 * M_2 * M_3 * M_4$$

$$= \delta_{16}[2,4,4,2,2,4,4,2,3,7,1,5,3,7,1,5,\\ 1,7,3,5,1,7,3,5,4,8,2,6,4,8,2,6,\\ 2,4,4,2,10,12,12,10,3,7,1,5,11,15,9,13,\\ 1,7,3,5,9,15,11,13,4,8,2,6,12,16,10,14]. \tag{8.4.19}$$

令

$$\Omega(0) = W_0 = \{C = (1,1,0,1)\}.$$

那么, $\Omega(0)$ 的真值矩阵为

$$T_{\Omega(0)} = \begin{bmatrix} 0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0 \\ 0&0&1&0&0&0&1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0 \end{bmatrix}. \tag{8.4.20}$$

从 (8.4.20) 可得

$$W_1 = \delta_{16}\{3,7,9,13\}. \tag{8.4.21}$$

因为 $(1,1,2,1) \sim \delta_{16}^3 \in W_1$, 故 X^d 是控制不动点.

然后设

$$\Omega(1) = W_1 \backslash W_0 = \delta_{16}\{7,9,13\}. \tag{8.4.22}$$

考察

$$(F_1(X,U), F_2(X,U), F_3(X,U), F_4(X,U)) \in \Omega(1), \tag{8.4.23}$$

可得 $\Omega(1)$ 的真值矩阵

$$T_{\Omega(1)} = \begin{bmatrix} 0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0 \\ 0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0&0&0&1&1 \\ 0&1&0&0&1&0&0&1&0&0&0&0&0&0&0&0 \end{bmatrix}. \tag{8.4.24}$$

于是有

$$W_2 = \delta_{16}\{2,5,6,8,10,14,15,16\}. \tag{8.4.25}$$

$$\Omega(2) = W_2 \backslash \{W_1 \cup W_0\} = \delta_{16}\{2,5,6,8,10,14,15,16\}. \tag{8.4.26}$$

第三步, 考察

$$(F_1(X,U),F_2(X,U),F_3(X,U),F_4(X,U))\in\Omega(2), \tag{8.4.27}$$

可得 $\Omega(2)$ 的真值矩阵

$$T_{\Omega(2)}=\begin{bmatrix}1&0&0&1&1&0&0&1&0&0&0&1&0&0&0&1\\0&0&0&1&0&0&0&1&0&1&1&1&0&1&1&1\\1&0&0&1&1&0&0&1&0&0&0&1&0&1&0&0\\0&0&0&1&0&1&0&0&0&1&1&1&0&1&1&1\end{bmatrix}. \tag{8.4.28}$$

于是有

$$W_3=\delta_{16}\{1,4,5,6,8,10,11,12,14,15,16\}, \tag{8.4.29}$$

以及

$$\Omega(3):=W_3\backslash\{W_2\cup W_1\cup W_0\}=\delta_{16}\{1,4,11,12\}. \tag{8.4.30}$$

现在, 因为

$$\Omega(0)\cup\Omega(1)\cup\Omega(2)\cup\Omega(3)=\varDelta_{16},$$

布尔控制网络 (8.4.17) 由状态反馈控制可镇定到 $(1,1,0,1)$.

最后, 我们构造状态反馈控制, 它必须满足以下条件

$$U|_{\Omega(1)\cup\Omega(0)}\leqslant T_{\Omega(0)}|_{\Omega(1)\cup\Omega(0)}, \tag{8.4.31a}$$

$$U|_{\Omega(2)}\leqslant T_{\Omega(1)}|_{\Omega(2)}, \tag{8.4.31b}$$

$$U|_{\Omega(3)}\leqslant T_{\Omega(2)}|_{\Omega(3)}. \tag{8.4.31c}$$

对于 (8.4.31a), 一个合理选择是

$$\begin{aligned}&u(\delta_{16}^3)=\delta_4^4\in\delta_4\{2,4\},\\&u(\delta_{16}^7)=\delta_4^2,\\&u(\delta_{16}^9)=\delta_4^3\in\delta_4\{1,3\},\\&u(\delta_{16}^{13})=\delta_4^1.\end{aligned} \tag{8.4.32}$$

注意, 这里记号 $\in\delta_4\{*,*\}$ 表示该值有多种选择, 花括号内为所有有效选择.

对于 (8.4.31b), 一个合理选择是

$$\begin{aligned}
u(\delta_{16}^2) &= \delta_4^2 \in \delta_4\{2,4\},\\
u(\delta_{16}^5) &= \delta_4^4,\\
u(\delta_{16}^6) &= \delta_4^2,\\
u(\delta_{16}^8) &= \delta_4^4,\\
u(\delta_{16}^{10}) &= \delta_4^3 \in \delta_4\{1,3\},\\
u(\delta_{16}^{14}) &= \delta_4^1,\\
u(\delta_{16}^{15}) &= \delta_4^3,\\
u(\delta_{16}^{16}) &= \delta_4^3.
\end{aligned} \tag{8.4.33}$$

对于 (8.4.31c), 一个合理选择是

$$\begin{aligned}
u(\delta_{16}^1) &= \delta_4^1 \in \delta_4\{1,3\},\\
u(\delta_{16}^4) &= \delta_4^2 \in \delta_4\{1,2,3,4\},\\
u(\delta_{16}^{11}) &= \delta_4^2 \in \delta_4\{2,4\},\\
u(\delta_{16}^{12}) &= \delta_4^4 \in \delta_4\{1,2,3,4\}.
\end{aligned} \tag{8.4.34}$$

将 (8.4.32)—(8.4.34) 放到一起, 即得一状态反馈镇定器如下:

$$u(t) = M_G x(t), \tag{8.4.35}$$

这里,

$$M_G = \delta_4[1,2,4,2,4,2,2,4,3,3,2,4,1,1,3,3].$$

于是,

$$\begin{aligned}
M_G^1 &= (I_2 \otimes \mathbf{1}_2^{\mathrm{T}})M_G\\
&= \delta_2[1,1,2,1,2,1,1,2,2,2,1,2,1,1,2,2];\\
M_G^2 &= (\mathbf{1}_2^{\mathrm{T}} \otimes I_2)M_G\\
&= \delta_2[1,2,2,2,2,2,2,2,1,1,2,2,1,1,1,1].
\end{aligned}$$

于是, 状态反馈镇定器的逻辑表达式如下:

$$\begin{aligned}
U_1(t) = &(X_1(t) \wedge X_2(t) \wedge X_3(t))\\
&\vee (X_1(t) \wedge X_2(t) \wedge \neg X_3(t) \wedge \neg X_4(t))\\
&\vee (X_1(t) \wedge \neg X_2(t) \wedge X_3(t) \wedge \neg X_4(t))
\end{aligned}$$

$$
\begin{aligned}
&\vee (X_1(t) \wedge \neg X_2(t) \wedge \neg X_3(t) \wedge X_4(t)) \\
&\vee (\neg X_1(t) \wedge X_2(t) \wedge \neg X_3(t) \wedge X_4(t)) \\
&\vee (\neg X_1(t) \wedge \neg X_2(t) \wedge X_3(t)) \qquad (8.4.36)\\
U_2(t) = &(X_1(t) \wedge X_2(t) \wedge X_3(t) \wedge X_4(t)) \\
&\vee (\neg X_1(t) \wedge X_2(t) \wedge X_3(t)) \\
&\vee (\neg X_1(t) \wedge X_2(t) \wedge X_3(t)).
\end{aligned}
$$

最后, 我们检验一下这个状态反馈镇定器. 将它代入布尔控制网络 (8.4.17) 后, 闭环系统的代数状态空间表达式为

$$
\begin{aligned}
x(t+1) &= M_F u(t) x(t) = M_F M_G x(t) x(t) \\
&= M_F M_G \, \mathrm{PR}_{16} \, x(t) := M_c x(t), \qquad (8.4.37)
\end{aligned}
$$

这里,

$$
\begin{aligned}
M_c &= M_F M_G \, \mathrm{PR}_{16} \\
&= \delta_{16}[2,7,3,5,9,7,3,13,3,7,2,6,3,7,9,13].
\end{aligned}
$$

不难检验

$$
M_c^3 = \delta_{16}[3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3].
$$

也就是说, 经过三步迭代后, 闭环系统收敛到 $\delta_{16}^3 \sim (1,1,2,1)$.

闭环系统 (8.4.37) 的状态转移图见图 8.4.1, 这里, 唯一的吸引子是 $\delta_{16}^3 \sim (1, 1, 0, 1)$.

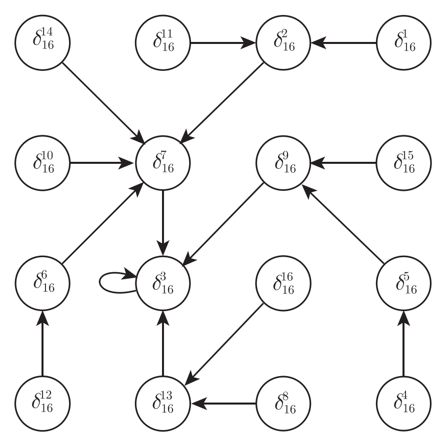


图 8.4.1　闭环系统 (8.4.37) 的状态转移图

最后, 值得一提的是, 从 (8.4.32)—(8.4.34) 可知, 这里有

$$2 \times 2 \times 2 \times 2 \times 2 \times 4 \times 2 \times 4 = 1024$$

个不同的 (时间最优) 状态反馈控制器.

8.5 集合镇定

定义 8.5.1 考察布尔控制网络 (8.2.2). 设 $W \subset \mathcal{D}^n$ 为一预先给定的状态集合. 该布尔控制网络称为可镇定到集合 W, 如果存在状态反馈控制 (8.3.1) 使得闭环网络收敛到 W. 即, 存在 $T > 0$ 使得 $X(t) \in W$, $t \geqslant T$.

定义 8.5.2 考察布尔控制网络 (8.2.2). 设 $W_0 \subset \mathcal{D}^n$. W_0 称为控制不变集, 如果对每个点 $X_0 \in W_0$ 存在一组控制 $U = (U_1, U_2, \cdots, U_m)$ 使得 $F(X_0, U) \in W_0$.

于是, 显然有如下的条件.

命题 8.5.1 布尔控制网络 (8.2.2) 可镇定到集合 W, 当且仅当, 该网络可镇定到 W_0, 这里

$$W_0 = \mathcal{I}(W)$$

为 W 中的最大控制不变集.

下面的结果用于检验一个集合是否为控制不变集.

命题 8.5.2 考察布尔控制网络 (8.2.2). 设 $W \subset \Delta^n$ 给定. 记相应于 W 的真值矩阵为 T_W. 由 T_W 决定的极大集为 W_0. 那么, W 是控制不变集, 当且仅当,

$$W \subset W_0. \tag{8.5.1}$$

证明 由定义, 至少存在一个 W_0 前提解 $U(X)$ 使得 $F(X, U(X)) \in W$, $X \in W_0$. 由于 $W \subset W_0$, 可知

$$F(X, U(X)) \in W, \quad X \in W. \qquad \square$$

类似于点镇定的情况, 可以构造集合镇定控制的设计方法, 它同时也是检验可否集合镇定的算法.

算法 8.5.1 第 0 步 (初始步): 设初始目标集为 W, 找出其最大控制不变集 $W_0 = \mathcal{I}(W)$. 设 $\Omega(0) = W_0$, 这里 $X^d = (C_1, C_2, \cdots, C_n) \in W_0$ 是系统收敛的目标. 构造 $T_{\Omega(0)}$, 利用它找到对应于 $\Omega(0)$ 的极大集 W_1. 然后

(i) 检验是否 $\Omega(0) \subset W_1$? 如果 $W_0 \not\subset W_1$, 则 W_0 不是控制不变集, 因此, 状态反馈镇定无解. 算法终止.

(ii) 如果 $W_0 \subset W_1$, 令 $k = 1$, 进入循环步骤.

第 k 步 (循环步骤, $k \geqslant 1$): 与算法 8.4.1 的第 k 步相同.

最后一步 (构造反馈镇定控制): 与算法 8.4.1 的最后一步相同.

类似于定理 8.4.2, 下面的定理也成立.

定理 8.5.1　布尔控制网络 (8.2.2) 可镇定到集合 W_0, 当且仅当, 算法 8.5.1 可以给出状态反馈镇定控制器.

以下例子给出一个对算法的解释.

例 8.5.1　回顾例 8.4.3. 考察布尔控制网络 (8.4.17), 令 $W=\{(1,2,1,2)\sim\delta_{16}^6, (1,2,2,1)\sim\delta_{16}^7, (2,1,2,1)\sim\delta_{16}^{12}\}$. 我们考察是否布尔控制网络 (8.4.17) 可被镇定到 W?

不难验证, $W_0=\mathcal{I}(W)=W$. 首先, 不妨检查一下, 系统可否被镇定到 W_0 中的某一个点, 如果可以, 就无须集合镇定了. 不难检验, $X_1=(1,2,1,2)$, $X_2=(1,2,2,1)$ 及 $X_3=(2,1,2,1)$ 都不是控制不动点, 因此, 不可能用点镇定代替集合镇定.

下面利用算法 8.5.1 进行集合镇定的检验和设计.

令 $\Omega(0)=W_0$, 则

$$
\begin{aligned}
T_{\Omega(0)}=\delta_{16}[&0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,\\
&0,1,0,0,0,1,0,0,0,0,0,1,0,0,0,1,\\
&0,0,0,0,0,1,1,0,0,1,0,0,0,0,0,0,\\
&0,1,0,0,0,0,0,0,0,0,0,1,1,0,0,0],
\end{aligned}
$$

极大集为

$$
W_1=\delta_{16}\{2,6,7,10,12,13,14,16\}.
$$

由于 $\Omega(0):=W_0\subset W_1$, W_0 是控制不变集.

令 $\Omega(1):=W_1\backslash W_0$.

不难算得

$$
\begin{aligned}
T_{\Omega(1)}=\delta_{16}[&1,0,0,1,1,0,0,1,0,0,0,0,0,0,0,0,\\
&0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,\\
&1,0,0,1,1,0,0,1,0,0,0,0,0,0,0,1,\\
&0,0,0,0,0,0,0,1,0,0,1,0,0,1,1,1],
\end{aligned}
$$

于是可得

$$
W_2=\delta_{16}\{1,4,5,8,11,14,15,16\}.
$$

令

$$
\Omega(2):=W_2\backslash(W_1\cup W_0)=\delta_{16}\{1,4,5,8,11,15\},
$$

即可算得

$$
\begin{aligned}
T_{\Omega(2)}=\delta_{16}[&0,1,1,0,0,1,1,0,0,0,1,1,0,0,1,1,\\
&1,0,0,1,1,0,0,1,1,1,0,0,1,1,0,0,\\
&0,1,1,0,0,0,0,0,0,0,1,1,1,1,0,0,\\
&1,0,0,1,0,1,1,0,1,1,0,0,0,0,0,0].
\end{aligned}
$$

不难看出

$$W_3=\delta_{16}\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16\},$$

并且,

$$\Omega(3)=W_3\backslash(W_2\cup W_1\cup W_0)=\delta_{16}\{3,9\}.$$

因为

$$\Omega(0)\cup\Omega(1)\cup\Omega(2)\cup\Omega(3)=\varDelta_{16},$$

可知布尔控制网络 (8.4.17) 可被镇定到 $W_0=W$.

最后, 设计状态反馈镇定器.

对于 $X\in\Omega(0)\cup\Omega(1)$, $U(X)$ 由 $T_{\Omega(0)}$ 决定. 选一组可行解如下:

$$
\begin{aligned}
&u(\delta_{16}^{2})=\delta_4^4\in\delta_4\{2,4\},\\
&u(\delta_{16}^{6})=\delta_4^3\in\delta_4\{2,3\},\\
&u(\delta_{16}^{7})=\delta_4^3,\\
&u(\delta_{16}^{10})=\delta_4^1\in\delta_4\{1,3\},\\
&u(\delta_{16}^{12})=\delta_4^4\in\delta_4\{2,4\},\\
&u(\delta_{16}^{14})=\delta_4^1,\\
&u(\delta_{16}^{16})=\delta_4^2.
\end{aligned}
\tag{8.5.2}
$$

利用 $T_{\Omega(1)}$, 可决定当 $X\in\Omega(2)$ 时的 $U(X)$, 选

$$
\begin{aligned}
&u(\delta_{16}^{1})=\delta_4^1\in\delta_4\{1,3\},\\
&u(\delta_{16}^{4})=\delta_4^1\in\delta_4\{1,3\},\\
&u(\delta_{16}^{8})=\delta_4^4\in\delta_4\{1,3,4\},\\
&u(\delta_{16}^{11})=\delta_4^2\in\delta_4\{2,4\},\\
&u(\delta_{16}^{15})=\delta_4^4\in\delta_4\{2,4\}.
\end{aligned}
\tag{8.5.3}
$$

利用 $T_{\Omega(2)}$, 当 $X\in\Omega(3)$ 时可选 $U(X)$ 如下:

$$\begin{aligned}u(\delta_{16}^3)&=\delta_4^3\in\delta_4\{1,3\},\\ u(\delta_{16}^9)&=\delta_4^2\in\delta_4\{2,4\}.\end{aligned}\tag{8.5.4}$$

记

$$u(t)=M_Gx(t),$$

这里, $M_G\in\mathcal{L}_{4\times16}$. 综合 (8.5.2)—(8.5.4), 一个可行的镇定器满足

$$M_G=\delta_4[1,2,3,1,1,3,3,4,2,1,2,4,4,1,4,2].\tag{8.5.5}$$

因此, (8.5.5) 给出了布尔控制网络 (8.4.17) 的一个状态反馈镇定器.

最后, 我们可以检验状态反馈控制下的闭环系统, 它是

$$\begin{aligned}x(t+1)&=M_Fu(t)x(t)=M_FM_Gx(t)x(t)\\ &=M_FM_G\,\mathrm{PR}_{16}\,x(t):=M_Cx(t).\end{aligned}\tag{8.5.6}$$

不难算出

$$M_C=\delta_{16}[2,7,4,2,2,12,12,13,4,7,2,6,12,7,10,6].\tag{8.5.7}$$

闭环系统 (8.5.6) 的状态转移图见图 8.5.1, 它唯一的吸引子是极限环 $C=\{\delta_{12}^6,\ \delta_{12}^{12}\}$. 因此, 所有的轨线均收敛于 $C\subset W_0=W$.

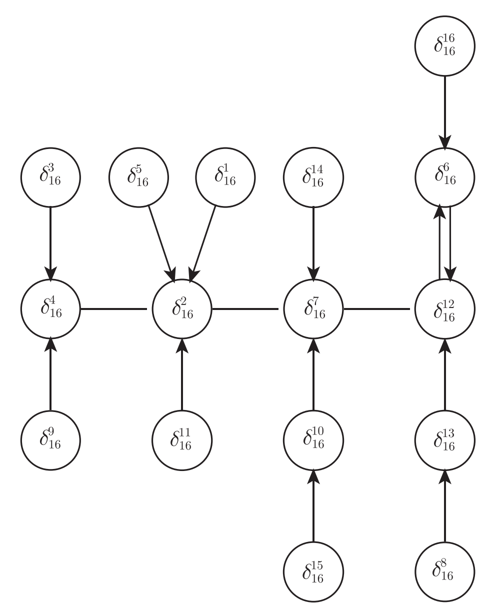


图 8.5.1　闭环系统 (8.5.6) 的状态转移图

由 (8.5.2)—(8.5.4) 可知, 有效的状态转移镇定器有 6144 个之多.

8.6　一般逻辑网络的镇定

前面两节对布尔控制网络状态反馈镇定器的设计方法, 对 k 值或混合值逻辑网络同样可用. 本节利用以下例子来说明这一点.

例 8.6.1　考察一个混合值逻辑控制网络

$$\begin{cases} X_1(t+1) = X_1(t) \Diamond X_2(t), \\ X_2(t+1) = X_1(t) \Box U(t), \end{cases} \tag{8.6.1}$$

这里 $X_1(t) \in \mathcal{D}_2$, $X_2(t)$, $U(t) \in \mathcal{D}_3$, $\Diamond : \mathcal{D}_2 \times \mathcal{D}_3 \to \mathcal{D}_2$, $\Box : \mathcal{D}_2 \times \mathcal{D}_3 \to \mathcal{D}_3$, 其相应的结构矩阵为

$$M_\Diamond = \delta_2[1,1,2,2,2,1]; \tag{8.6.2}$$

$$M_\Box = \delta_3[1,2,3,2,3,1]. \tag{8.6.3}$$

(i) 可否设计状态反馈控制, 使 (8.6.1) 镇定到 $(1,1) \sim \delta_6^1$?

容易得到网络 (8.6.1) 的代数状态空间表达式

$$x(t+1) = M_F u(t) x(t), \tag{8.6.4}$$

这里,

$$M_F = \delta_6[1,1,4,5,5,2,2,2,5,6,6,3,3,3,6,4,4,1].$$

令 $\Omega(0) = W_0 = \{(1,1)\}$. 利用 (8.6.4) 可得系统关于容许集合 $\Omega(0)$ 的真值矩阵

$$T_{\Omega(0)} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{8.6.5}$$

因此, 可得极大集

$$W_1 = \delta_6\{1,2,6\}. \tag{8.6.6}$$

然后定义

$$\Omega(1) = W_1 \backslash W_0 = \delta_6\{2,6\}. \tag{8.6.7}$$

系统关于 $\Omega(1)$ 的真值矩阵为

$$T_{\Omega(1)} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{8.6.8}$$

从而可得

$$W_2 = \delta_6\{1, 2, 3, 4, 5, 6\} \tag{8.6.9}$$

且

$$\Omega(2) = W_2 \backslash (W_0 \cup W_1) = \delta_6\{3, 4, 5\}. \tag{8.6.10}$$

现在

$$\Omega(0) \cup \Omega(1) \cup \Omega(2) = \mathcal{D}_6,$$

因此, 混合值逻辑控制网络 (8.6.1) 可以被反馈镇定到 $(1, 1) \sim \delta_6^1$.

下面构造反馈镇定器. 与布尔控制网络类似, 可以在子集 $\Omega(0)$, $\Omega(1)$, $\Omega(2)$ 上逐次构造可行的反馈控制略去过程, 一个可行状态反馈控制为

$$u(t) = M_G x(t), \tag{8.6.11}$$

这里,

$$M_G = \delta_3[1, 1, 3, 2, 2, 3].$$

为检验这个结果, 写出反馈后的闭环系统

$$x(t+1) = M_F M_G x^2(t) = M_F M_G \, \mathrm{PR}_6 \, x(t) := M_c x(t), \tag{8.6.12}$$

于是

$$M_c = M_F M_G \, \mathrm{PR}_6 = \delta_6[1, 1, 6, 6, 6, 1].$$

因为

$$M_c^2 = \delta_6[1, 1, 1, 1, 1, 1],$$

故经两次迭代, 网络即收敛到 δ_6^1.

(ii) 可否设计状态反馈控制, 使 (8.6.1) 镇定到 $W = \{(1, 3), (2, 3)\}$?

首先, 可以证明 $W_0 = \mathcal{I}(W) = W$.

其次, 不难验证, $(1, 3) \sim \delta_6^3$ 和 $(2, 3) \sim \delta_6^6$ 都不是控制不动点, 因此, 不可能用点镇定来实现集合镇定.

下面考虑集合镇定. 令 $\Omega(0)=W_0$. 则有关于容许集 $\Omega(0)$ 的真值矩阵

$$T_{\Omega(0)}=\begin{bmatrix}0&0&0&0&0&0\\0&0&0&1&1&1\\1&1&1&0&0&0\end{bmatrix}. \tag{8.6.13}$$

因此, 可行控制为

$$u(t)=\delta_3[3,3,3,2,2,2]x(t). \tag{8.6.14}$$

它可将系统镇定到 W_0.

为验证这一点, 将该控制代入原系统, 得到闭环系统

$$x(t+1)=M_FM_G\,\mathrm{PR}_6\,x(t):=M_cx(t), \tag{8.6.15}$$

这里,

$$M_c=M_FM_G\,\mathrm{PR}_6=\delta_6[3,3,6,6,6,3].$$

图 8.6.1 是闭环系统 (8.6.15) 的状态转移图. 从图中可看出系统到 W_0 的收敛性.

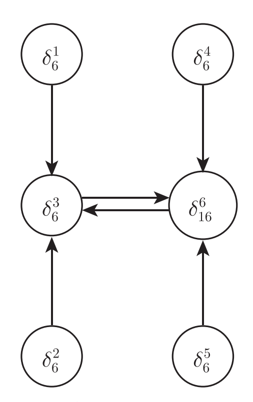


图 8.6.1 闭环系统 (8.6.15) 的状态转移图

第 9 章　逻辑系统辨识

辨识问题是动态系统研究中的一个带有根本性的问题. 因为人们观察到的只是数据, "辨识" 就是要从观测到的数据推断出系统模型. 这对于动态系统的分析与控制都极具重要性.

布尔网络的原始模型来自基因调控网络, 因此, 其辨识问题具有更重要的实际背景. 这方面的讨论很多, 例如见文献 [106] 及其中的参考文献. 利用矩阵半张量积方法探讨布尔 (控制) 网络的辨识的文章包括文献 [19, 20, 26, 46] 等. 本章所介绍的是一种新的方法, 更详尽的讨论可见文献 [118].

9.1　布尔网络的辨识

9.1.1　辨识的唯一性问题

考察一个布尔网络

$$\begin{cases} X_1(t+1)=F_1(X_1(t),X_2(t),\cdots,X_n(t)),\\ X_2(t+1)=F_2(X_1(t),X_2(t),\cdots,X_n(t)),\\ \qquad\qquad\cdots\cdots\\ X_n(t+1)=F_n(X_1(t),X_2(t),\cdots,X_n(t)),\end{cases}\\ Y_j(t)=C_j(X_1(t),X_2(t),\cdots,X_n(t)),\quad j=1,2,\cdots,p, \tag{9.1.1}$$

这里 $X_i(t)\in\mathcal{D}$, $i=1,2,\cdots,n$ 为状态变量, $Y_j(t)\in\mathcal{D}$, $j=1,2,\cdots,p$ 为观测变量, $F_i:\mathcal{D}^n\to\mathcal{D}$, $C_j:\mathcal{D}^p\to\mathcal{D}$ 为布尔函数.

网络 (9.1.1) 的代数状态空间表示为

$$\begin{cases} x(t+1)=Mx(t),\\ y(t)=Hx(t),\end{cases} \tag{9.1.2}$$

这里, $x(t)=\ltimes_{i=1}^n x_i(t)$, $y(t)=\ltimes_{j=1}^p y_j(t)$; $x_i(t)=\vec{X}_i(t)$, $y_j(t)=\vec{Y}_j(t)$ 分别为 $X_i(t)$ 和 $Y_j(t)$ 的向量表示; $M\in\mathcal{L}_{2^n\times 2^n}$, $H\in\mathcal{L}_{2^p\times 2^n}$. 称 (9.1.1) 为带观测输出的布尔网络.

定义 9.1.1 给定布尔网络 (9.1.1) 和一组观测数据

$$O_Y = \{Y(t), \mid t \in J \subset \mathbb{Z}_+ = \{0, 1, 2, \cdots\}\}. \tag{9.1.3}$$

由 O_Y 来确定布尔函数 F_i, $i = 1, 2, \cdots, n$ 和 C_j, $j = 1, 2, \cdots, p$ 称为布尔网络的辨识问题.

因为布尔 (控制) 网络同它的代数状态空间表达式是等价的, 而直接辨识逻辑函数是很困难的, 因此, 下面讨论的布尔网络的辨识问题是指利用 O_Y 确定 (9.1.2) 中的 M 与 H.

我们首先关心的问题是: (9.1.2) 中的 M 与 H 能否唯一确定.

设 $z(t) = Tx(t)$, 这里 $T \in \mathcal{L}_{2^n \times 2^n}$ 非奇异. 因此, $X(t) \to Z(t)$ 是一个坐标变换. 那么, 在 Z 坐标下, 系统 (9.1.2) 变为

$$\begin{cases} z(t+1) = \tilde{M} z(t), \\ y(t) = \tilde{H} z(t), \end{cases} \tag{9.1.4}$$

这里

$$\tilde{M} = TMT^{\mathrm{T}}, \quad \tilde{H} = HT^{\mathrm{T}}. \tag{9.1.5}$$

如果 $z(0) = Tx(0)$, 那么, 输出序列是完全相同的. 因此, 任何 O_Y 都不可能区分 (M, H) 与 $(\tilde{M}, \tilde{H})$. 实际上, 区分它们也是没有必要的, 因为它们是系统内部的结构, 完全不影响状态到输出的映射关系.

(M, H) 与 $(\tilde{M}, \tilde{H})$ 称为等价的, 如果它们满足 (9.1.5). 因此, 布尔网络的辨识问题指的是找出这等价类中的任何一个.

9.1.2 布尔网络辨识的要求与算法

定义 9.1.2 带有观测输出的布尔网络 (9.1.1) 称为可观测的, 如果对任何两个不同的初值 $X(0) \neq \tilde{X}(0)$, 其相应的输出序列不相等, 即

$$(Y(0), Y(1), \cdots) \neq (\tilde{Y}(0), \tilde{Y}(1), \cdots).$$

下面给出 (9.1.1) 可辨识的一个预设条件.

假设 9.1 布尔网络是可观测的.

定义 9.1.3 考察布尔网络 (9.1.1). 令 $X(0) = a$, 那么, 相应的输出序列, 记为

$$Y^a := (Y(0), Y(1), \cdots),$$

称为全输出序列. 其前 2^n+1 个元素, 记作

$$Y_e^a := (Y(0), Y(1), \cdots, Y(2^n)),$$

称为有效输出序列.

命题 9.1.1　全输出序列是由有效输出序列决定的.

证明　情况一: 设 $(X(0), X(1), \cdots, X(2^n-1))$ 中没有重复元素, 那么, 由于状态空间 $|\mathcal{D}^n| = 2^n$, 则必有 $X(i)$, $0 \leqslant i \leqslant 2^n-1$, 使得 $X(2^n) = X(i)$, 于是 $X(2^n+1) = X(i+1)$, 从而 $Y(2^n+1) = Y(i+1)$. 以此类推, 所有的 $Y(t)$, $t > 2^n$ 均唯一确定.

情况二: 设存在 $0 \leqslant i < j \leqslant 2^n-1$, 使 $X(i) = X(j)$. 那么 $(X(i), X(i+1), \cdots, X(j) = X(i))$ 就是一个极限环. 此后的元素都在环内. 故存在 $X(k)$, $i \leqslant k < j$, 使 $X(2^n) = X(k)$. 类似情况一可知, $X(2^n)$ 以及所有的 $Y(t)$, $t > 2^n$ 均唯一确定. □

推论 9.1.1　如果布尔网络 (9.1.1) 是可观测的, 则对任何两个初值点 $X(0) \neq \tilde{X}(0)$, 它们生成的有效输出序列是不同的. 因此, 对所有 2^n 个初值点, 共有 2^n 组不同的有效输出序列.

一个显然的事实是, 一个系统要能辨识必须有足够的观测数据. 那么, 对于布尔网络 (9.1.1) 多少数据才是足够多呢? 观测数据 O_Y 称为完备的, 如果它包含所有的有效输出序列. 由推论 9.1.1 可知, 当网络可观时, 共有 2^n 组不同的有效输出序列.

下面给出主要可辨识定理.

定理 9.1.1　考察布尔网络 (9.1.1). 设此网络可观测, 那么, 该网络是在完备观测数据下可辨识的.

下面这个算法实际上给出了这个定理的一个构造性证明.

算法 9.1.1　第一步: 找出完备观测数据中的所有有效输出序列. 排成 (顺序可以任意)

$$\begin{aligned} Y_e^1 &= (y^1(0), y^1(1), \cdots, y^1(2^n)), \\ Y_e^2 &= (y^2(0), y^2(1), \cdots, y^2(2^n)), \\ &\cdots\cdots \\ Y_e^{2^n} &= (y^{2^n}(0), y^{2^n}(1), \cdots, y^{2^n}(2^n)). \end{aligned} \tag{9.1.6}$$

第二步: 构造状态转移矩阵 M: Y_e^j 称为 Y_e^i 的后继 (或 Y_e^i 称为 Y_e^j 的前身), 记作 $Y_e^j \succ Y_e^i$ (相应地, $Y_e^i \prec Y_e^j$), 如果

$$y^j(t) = y^i(t+1), \quad t = 0, 1, \cdots, 2^n. \tag{9.1.7}$$

对每个 Y_e^i 找出它的后继 Y_e^j. 然后定义

$$\mathrm{Col}_i(M) = \delta_{2^n}^j, \quad i = 1, \cdots, 2^n, \tag{9.1.8}$$

即得状态转移矩阵 M.

第三步: 构造输出矩阵 H:

$$H = \left[y^1(0), y^2(0), \cdots, y^{2^n}(0)\right]. \tag{9.1.9}$$

注 9.1.1 关于算法 9.1.1 的几点说明:

(i) 注意, (9.1.7) 中的 $y^i(2^n+1)$ 是根据 Y_e^i 推算出来的.

(ii) 根据 (9.1.7) 构造出来的是唯一的有效输出序列, 根据观测数据的完备性, 每个有效输出序列必有且仅有一个后继. 因此, 假设条件保证了算法的可实现性.

(iii) 实际上, 我们假定 Y_e^i 是由 $x^i(0) = \delta_{2^n}^i$ 生成的, $i = 1, 2, \cdots, 2^n$. 这就是 (9.1.8) 的构造原理. 为什么可以将 $x^i(0)$ 设为 $\delta_{2^n}^i$ 呢? 这来自等价性. 换一种排序无非就是一个坐标变换.

(iv) 因为

$$y^i(0) = Hx^i(0) = H\delta_{2^n}^i = \mathrm{Col}_i(H),$$

即得 (9.1.9).

注 9.1.2 算法 9.1.1 说明了定理 9.1.1 的正确性. 其实, 从算法不难看出, 在网络可观的假定下, 观测数据的完备性也是必要的. 因为如果观测数据少了某组有效输出序列, 那么, 这组序列所对应的变量是谁的后继、谁的前身均无从判定.

下面给出一个例子.

例 9.1.1 考察如下系统:

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), X_3(t), X_4(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), X_3(t), X_4(t)), \\ X_3(t+1) = F_3(X_1(t), X_2(t), X_3(t), X_4(t)), \\ X_4(t+1) = F_4(X_1(t), X_2(t), X_3(t), X_4(t)), \end{cases}$$
$$\begin{aligned} &Y_1(t) = C_1(F_4(X_1(t), X_2(t), X_3(t), X_4(t))), \\ &Y_2(t) = C_2(F_4(X_1(t), X_2(t), X_3(t), X_4(t))), \\ &Y_3(t) = C_3(F_4(X_1(t), X_2(t), X_3(t), X_4(t))), \end{aligned} \tag{9.1.10}$$

这里 $X_1(t), X_2(t), X_3(t), X_4(t), Y_1(t), Y_2(t), Y_3(t) \in \mathcal{D}$, $F_i : \mathcal{D}^4 \to \mathcal{D}$, $i = 1, 2, 3, 4$, $h_j : \mathcal{D}^4 \to \mathcal{D}$, $j = 1, 2, 3$ 为布尔函数. 记 $x_i(t) = \vec{X}_i(t)$, $y_j(t) = \vec{Y}_j(t)$, $x(t) =$

$\ltimes_{i=1}^4 x_i(t)$, $y(t) = \ltimes_{j=1}^3 y_j(t)$, 则可记系统 (9.1.10) 的代数状态空间表达式为

$$\begin{aligned} x(t+1) &= Mx(t), \\ y(t) &= Hx(t). \end{aligned} \tag{9.1.11}$$

现在假定一组完备的观测数据如下:

$$\begin{aligned}
Y_e^1 &= \delta_8\{1,7,7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,[7]\}, \\
Y_e^2 &= \delta_8\{3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,[7]\}, \\
Y_e^3 &= \delta_8\{7,7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,[7]\}, \\
Y_e^4 &= \delta_8\{5,7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,[7]\}, \\
Y_e^5 &= \delta_8\{5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,5,[3]\}, \\
Y_e^6 &= \delta_8\{7,3,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,[7]\}, \\
Y_e^7 &= \delta_8\{7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,[5]\}, \\
Y_e^8 &= \delta_8\{5,6,5,6,5,6,5,6,5,6,5,6,5,6,5,6,5,[6]\}, \\
Y_e^9 &= \delta_8\{1,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,[5]\}, \\
Y_e^{10} &= \delta_8\{3,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,[5]\}, \\
Y_e^{11} &= \delta_8\{7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,[5]\}, \\
Y_e^{12} &= \delta_8\{5,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,7,[5]\}, \\
Y_e^{13} &= \delta_8\{6,5,6,5,6,5,6,5,6,5,6,5,6,5,6,5,6,[5]\}, \\
Y_e^{14} &= \delta_8\{8,6,8,7,5,3,7,7,5,3,7,7,5,3,7,7,5,[3]\}, \\
Y_e^{15} &= \delta_8\{8,7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,7,[7]\}, \\
Y_e^{16} &= \delta_8\{6,8,7,5,3,7,7,5,3,7,7,5,3,7,7,5,3,[7]\}.
\end{aligned} \tag{9.1.12}$$

注意: 这里括号内的最后一个数 $[a]$ 表示不是观测数据, 它是推算出来的, 用以寻找后继数.

我们希望从观测数据 (9.1.12) 辨识出 M 和 H.

不妨设生成 Y_e^i 的初始值为 $X_i(0) = \delta_{16}^i$, 即 $Y_e^i \Rightarrow X_i(0) = \delta_{16}^i$, $i = 1,2,\cdots,16$. 根据式 (9.1.7) 检验 (9.1.12) 中的有效输出序列, 容易看出

$$\begin{aligned}
&\delta_{16}^1 \succ \delta_{16}^3 \succ \delta_{16}^7 \succ \delta_{16}^5 \succ \delta_{16}^2 \succ \delta_{16}^3; \\
&\delta_{16}^4 \succ \delta_{16}^7; \\
&\delta_{16}^6 \succ \delta_{16}^{10} \succ \delta_{16}^5; \\
&\delta_{16}^8 \succ \delta_{16}^{13} \succ \delta_{16}^8;
\end{aligned} \tag{9.1.13}$$

$$\delta_{16}^{9} \succ \delta_{16}^{5};$$
$$\delta_{16}^{11} \succ \delta_{16}^{5};$$
$$\delta_{16}^{12} \succ \delta_{16}^{5};$$
$$\delta_{16}^{14} \succ \delta_{16}^{16} \succ \delta_{16}^{15} \succ \delta_{16}^{7}.$$

根据式 (9.1.8) 即可得到

$$M = \delta_{16}[3,3,7,7,2,10,5,13,5,5,5,5,8,16,7,15]. \tag{9.1.14}$$

利用式 (9.1.9) 可得

$$H = \delta_{8}[1,3,7,5,5,7,7,5,1,3,7,5,6,8,8,6]. \tag{9.1.15}$$

最后, 不难找出, 这个系统为

$$\begin{cases} X_1(t+1) = X_2(t) \vee X_4(t), \\ X_2(t+1) = X_3(t) \wedge X_1(t), \\ X_3(t+1) = X_2(t) \bar{\vee} X_1(t), \\ X_4(t+1) = X_3(t) \to X_2(t), \end{cases}$$
$$\begin{aligned} &Y_1(t) = X_2(t) \wedge X_3(t), \\ &Y_2(t) = X_3(t) \leftrightarrow X_4(t), \\ &Y_3(t) = X_1(t) \vee X_2(t). \end{aligned} \tag{9.1.16}$$

下面讨论一下假设 9.1. 是不是不完全能观的系统就一定不能辨识呢? 我们考察下面这个例子.

例 9.1.2 考察布尔网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t)), \end{cases}$$
$$Y(t) = C(X(t)), \tag{9.1.17}$$

设其代数状态空间表达式为

$$\begin{cases} x(t+1) = Mx(t), \\ y(t) = Hx(t). \end{cases} \tag{9.1.18}$$

为分析, 暂时假定 M 与 H 已知, 设

$$M=\delta_4[1,1,2,2];\quad H=\delta_2[2,1,2,2]. \tag{9.1.19}$$

则由 $X^i(0)=\delta_4^i,\ i=1,2,3,4$ 生成的输出序列分别为

$$\begin{aligned}
&Y_O^1=(\delta_2^2,\delta_2^2,\delta_2^2,\delta_2^2,\cdots),\quad Y_O^2=(\delta_2^1,\delta_2^2,\delta_2^2,\delta_2^2,\cdots),\\
&Y_O^3=(\delta_2^2,\delta_2^1,\delta_2^2,\delta_2^2,\cdots),\quad Y_O^4=(\delta_2^2,\delta_2^1,\delta_2^2,\delta_2^2,\cdots).
\end{aligned} \tag{9.1.20}$$

显然, 系统是不可观测的, 因为 δ_4^3 与 δ_4^3 不可区分.

这时, 不难发现

$$\begin{aligned}
&\delta_4^2\succ\delta_4^1,\\
&\delta_4^3\succ\delta_4^2,\\
&\delta_4^4\succ\delta_4^2.
\end{aligned}$$

系统 (9.1.17) 的状态转移图如图 9.1.1 所示. 因为两个不可分点处于状态转移图末端, 它们没有前身, 因此, 实际上这个系统还是可以辨识的.

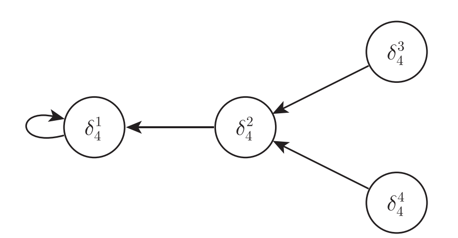


图 9.1.1　系统 (9.1.17) 的状态转移图

9.2　布尔控制网络的辨识

一个布尔控制网络可表示为

$$\begin{cases}
X_1(t+1)=F_1(X_1(t),X_2(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)),\\
X_2(t+1)=F_2(X_1(t),X_2(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)),\\
\qquad\qquad\cdots\cdots\\
X_n(t+1)=F_n(X_1(t),X_2(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)),
\end{cases}$$
$$Y_j(t)=C_j(X_1(t),X_2(t),\cdots,X_n(t)),\quad j=1,2,\cdots,p, \tag{9.2.1}$$

这里 $U_\alpha(t)\in\mathcal{D},\ \alpha=1,2,\cdots,m$ 为控制变量, $F_i:\mathcal{D}^{n+m}\to\mathcal{D}$ 为布尔函数, 其余同 (9.1.1). 网络 (9.2.1) 的代数状态空间表示为

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{9.2.2}$$

这里, $u(t) = \ltimes_{\alpha=1}^{m} u_\alpha(t)$, $u_\alpha(t) = \vec{U}_\alpha(t)$, $L \in \mathcal{L}_{2^{n+m} \times 2^n}$, 其余同 (9.1.2).

定义 9.2.1 给定布尔控制网络 (9.2.1) 和一组输入输出数据

$$O_{UY} = \{(U(t), Y(t)) \mid t \in J \subset \mathbb{Z}_+ = \{0, 1, 2, \cdots\}\}. \tag{9.2.3}$$

由 O_{UY} 来确定布尔函数 F_i, $i = 1, 2, \cdots, n$ 和 C_j, $j = 1, 2, \cdots, p$ 称为布尔控制网络的辨识问题.

在 Z 坐标下, 系统 (9.2.1) 变为

$$\begin{cases} z(t+1) = \tilde{L}u(t)z(t), \\ y(t) = \tilde{H}z(t), \end{cases} \tag{9.2.4}$$

这里

$$\tilde{L} = TL\left(I_{2^m} \otimes T^{\mathrm{T}}\right); \quad \tilde{H} = HT^{\mathrm{T}}. \tag{9.2.5}$$

如果 $z(0) = Tx(0)$, 那么, 对任何输入序列 $U(0), U(1), \cdots$, 相应的输出序列 $Y(0), Y(1), \cdots$ 也是完全相同的. 因此, 任何 O_{UY} 都不可能区分 (L, H) 与 $(\tilde{L}, \tilde{H})$. 同样, 区分它们也是没有必要的, 因为它们不影响状态到输出的映射关系.

(L, H) 与 $(\tilde{L}, \tilde{H})$ 称为等价的, 如果它们满足 (9.2.5). 因此, 布尔控制网络的辨识问题也指的是找出这等价类中的任何一个.

对于布尔控制网络的辨识, 最直观的想法是, 令控制为定常值, 则布尔控制网络就转化为布尔网络了. 即, 记

$$L = [L_1, L_2, \cdots, L_{2^m}],$$

那么, 当 $u(t) = \delta_{2^m}^i$ 时, (9.2.2) 变为子系统 (记作 Σ_i), 其方程为

$$\begin{cases} x(t+1) = L_i x(t), \\ y(t) = Hx(t), \quad i = 1, 2, \cdots, 2^m. \end{cases} \tag{9.2.6}$$

那么, 对每一块子系统将 L_i, H 辨识出来不就行了吗? 这里有一个困难, 因为我们只能得到每组 (L_i, H) 等价类中的一个, 因此, 各组数据得出的 H 就可能不同. 当然, 将各组 H 调成一样不是难事, 但即使 H 相同, 也不能说明 $\{L_i \mid i = 1, 2, \cdots, 2^m\}$ 相容. 即, 它们拼接出来的 L 与原系统一般不满足 (9.2.5).

定义 9.2.2 (i) 设 Y_e^s 为系统 (9.2.6) 中 $i=s$ 时的有效输出序列, $(Y_e^1, Y_e^2, \cdots, Y_e^{2^m})$ 称为一个相容有效输出序列集, 如果它们是由同一个 $X(0)$ 生成的.

(ii) 对于布尔控制网络, 一组观测数据称为完备的, 如果它包含所有相容有效观测序列集.

注 9.2.1 (i) 如果至少有一个子系统 Σ_i 是能观的, 那么布尔控制网络 (9.2.1) 就有 2^n 个相容有效输出序列集.

(ii) 对于一般的布尔控制网络, 得到完备观测数据不显然. 如果需要, 可通过构造观测器得到.

(iii) 假定网络是可控的, 则在一个子系统被辨识出来后, 可让系统控制到每个初态, 再从该初态出发, 观测其他各组控制下的输出数据, 这样, 可以得到相容有效输出序列集.

类似于布尔网络, 我们有如下定理.

定理 9.2.1 设布尔控制网络 (9.2.1) 子系统 Σ_i 是能观的, 那么, (9.2.1) 可由一组完备观测数据辨识出来.

类似于布尔网络的情况, 下面的算法给出一个构造性的证明.

算法 9.2.1 第一步: 找出完备观测数据中的所有相容有效输出序列集. 排成 (顺序可以任意)

$$\begin{array}{cccc} Y_e^1(\Sigma_1) & Y_e^1(\Sigma_2) & \cdots & Y_e^1(\Sigma_{2^m}) \\ Y_e^2(\Sigma_1) & Y_e^2(\Sigma_2) & \cdots & Y_e^2(\Sigma_{2^m}) \\ \vdots & \vdots & & \vdots \\ Y_e^{2^n}(\Sigma_1) & Y_e^{2^n}(\Sigma_2) & \cdots & Y_e^{2^n}(\Sigma_{2^m}), \end{array} \tag{9.2.7}$$

这里

$$Y_e^i(\Sigma_j) = (Y_{\Sigma_j}^i(0), Y_{\Sigma_j}^i(1), \cdots, Y_{\Sigma_j}^i(2^n)),$$

第二步: 设 Σ_k 可观, 则取 (9.2.7) 中的第 j 列

$$\begin{array}{c} Y_e^1(\Sigma_j) \\ Y_e^2(\Sigma_j) \\ \vdots \\ Y^{2^n}(\Sigma_j) \end{array}$$

利用算法 9.1.1 辨识出 L_j 和 H.

第三步: 分别取 (9.2.7) 中的第 i 列, $i \neq j$, 利用算法 9.1.1 辨识出 L_i.

下面给出一个例子.

例 9.2.1 考察如下布尔控制网络:

$$\begin{cases} X_1(t+1)=F_1(X_1(t),X_2(t),X_3(t),X_4(t),U(t)),\\ X_2(t+1)=F_2(X_1(t),X_2(t),X_3(t),X_4(t),U(t)),\\ X_3(t+1)=F_3(X_1(t),X_2(t),X_3(t),X_4(t),U(t)),\\ X_4(t+1)=F_4(X_1(t),X_2(t),X_3(t),X_4(t),U(t)), \end{cases}$$
$$\begin{aligned} &Y_1(t)=C_1(F_4(X_1(t),X_2(t),X_3(t),X_4(t))),\\ &Y_2(t)=C_2(F_4(X_1(t),X_2(t),X_3(t),X_4(t))),\\ &Y_3(t)=C_3(F_4(X_1(t),X_2(t),X_3(t),X_4(t))), \end{aligned} \tag{9.2.8}$$

这里 $X_1(t),X_2(t),X_3(t),X_4(t),U(t),Y_1(t),Y_2(t),Y_3(t)\in\mathcal{D}$, $F_i:\mathcal{D}^5\to\mathcal{D}$, $i=1,2,3,4$, $C_j:\mathcal{D}^4\to\mathcal{D}$, $j=1,2,3$ 为布尔函数. 记 $x_i(t)=\vec{X}_i(t)$, $u(t)=\vec{U}(t)$, $y_j(t)=\vec{Y}_j(t)$, $x(t)=\ltimes_{i=1}^4 x_i(t)$, $y(t)=\ltimes_{j=1}^3 y_j(t)$, 则可记系统 (9.1.10) 的代数状态空间表达式为

$$\begin{aligned} &x(t+1)=Lu(t)x(t),\\ &y(t)=Hx(t). \end{aligned} \tag{9.2.9}$$

现在假定一组完备的相容有效输出序列集如下:

$$\begin{matrix} Y_e^1(\Sigma_1) & Y_e^1(\Sigma_2)\\ Y_e^2(\Sigma_1) & Y_e^2(\Sigma_2)\\ \vdots & \vdots\\ Y_e^{16}(\Sigma_1) & Y_e^{16}(\Sigma_2), \end{matrix} \tag{9.2.10}$$

这里

$$\begin{aligned} &Y_e^1(\Sigma_1)=\delta_8\{2,7,2,7,2,7,2,7,2,7,2,7,2,7,2,7,2,[7]\},\\ &Y_e^2(\Sigma_1)=\delta_8\{2,6,2,6,2,6,2,6,2,6,2,6,2,6,2,6,2,[6]\},\\ &Y_e^3(\Sigma_1)=\delta_8\{1,3,6,6,1,3,6,6,1,3,6,6,1,3,6,6,1,[3]\},\\ &Y_e^4(\Sigma_1)=\delta_8\{3,6,6,1,3,6,6,1,3,6,6,1,3,6,6,1,3,[6]\},\\ &Y_e^5(\Sigma_1)=\delta_8\{6,1,3,6,6,1,3,6,6,1,3,6,6,1,3,6,6,[1]\},\\ &Y_e^6(\Sigma_1)=\delta_8\{6,6,1,3,6,6,1,3,6,6,1,3,6,6,1,3,6,[6]\},\\ &Y_e^7(\Sigma_1)=\delta_8\{5,5,1,3,6,6,1,3,6,6,1,3,6,6,1,3,6,[6]\},\\ &Y_e^8(\Sigma_1)=\delta_8\{7,6,3,6,6,1,3,6,6,1,3,6,6,1,3,6,6,[1]\},\\ &Y_e^9(\Sigma_1)=\delta_8\{6,5,1,3,6,6,1,3,6,6,1,3,6,6,1,3,6,[6]\}, \end{aligned} \tag{9.2.11}$$

$$Y_e^{10}(\Sigma_1)=\delta_8\{6,6,5,1,3,6,6,1,3,6,6,1,3,6,6,1,3,[6]\},$$
$$Y_e^{11}(\Sigma_1)=\delta_8\{5,1,3,6,6,1,3,6,6,1,3,6,6,1,3,6,6,[1]\},$$
$$Y_e^{12}(\Sigma_1)=\delta_8\{7,2,7,2,7,2,7,2,7,2,7,2,7,2,7,2,7,[2]\},$$
$$Y_e^{13}(\Sigma_1)=\delta_8\{6,3,6,6,1,3,6,6,1,3,6,6,1,3,6,6,1,[3]\},$$
$$Y_e^{14}(\Sigma_1)=\delta_8\{6,2,6,2,6,2,6,2,6,2,6,2,6,2,6,2,6,[2]\},$$
$$Y_e^{15}(\Sigma_1)=\delta_8\{5,7,2,7,2,7,2,7,2,7,2,7,2,7,2,7,2,[7]\},$$
$$Y_e^{16}(\Sigma_1)=\delta_8\{7,6,6,5,1,3,6,6,1,3,6,6,1,3,6,6,1,[3]\}.$$

$$\begin{aligned}
Y_e^{1}(\Sigma_2)&=\delta_8\{2,6,6,5,1,2,7,6,6,5,1,2,7,6,6,5,1,[2]\},\\
Y_e^{2}(\Sigma_2)&=\delta_8\{2,7,6,6,5,1,2,7,6,6,5,1,2,7,6,6,5,[1]\},\\
Y_e^{3}(\Sigma_2)&=\delta_8\{1,2,7,6,6,5,1,2,7,6,6,5,1,2,7,6,6,[5]\},\\
Y_e^{4}(\Sigma_2)&=\delta_8\{3,7,5,7,2,6,6,5,1,2,7,6,6,5,1,2,7,[6]\},\\
Y_e^{5}(\Sigma_2)&=\delta_8\{6,2,6,6,5,1,2,7,6,6,5,1,2,7,6,6,5,[1]\},\\
Y_e^{6}(\Sigma_2)&=\delta_8\{6,5,6,5,1,2,7,6,6,5,1,2,7,6,6,5,1,[2]\},\\
Y_e^{7}(\Sigma_2)&=\delta_8\{5,6,5,1,2,7,6,6,5,1,2,7,6,6,5,1,2,[7]\},\\
Y_e^{8}(\Sigma_2)&=\delta_8\{7,5,7,2,6,6,5,1,2,7,6,6,5,1,2,7,6,[6]\},\\
Y_e^{9}(\Sigma_2)&=\delta_8\{6,5,1,2,7,6,6,5,1,2,7,6,6,5,1,2,7,[6]\},\\
Y_e^{10}(\Sigma_2)&=\delta_8\{6,6,5,1,2,7,6,6,5,1,2,7,6,6,5,1,2,[7]\},\\
Y_e^{11}(\Sigma_2)&=\delta_8\{5,1,2,7,6,6,5,1,2,7,6,6,5,1,2,7,6,[6]\},\\
Y_e^{12}(\Sigma_2)&=\delta_8\{7,2,6,6,5,1,2,7,6,6,5,1,2,7,6,6,5,[1]\},\\
Y_e^{13}(\Sigma_2)&=\delta_8\{6,3,7,5,7,2,6,6,5,1,2,7,6,6,5,1,2,[7]\},\\
Y_e^{14}(\Sigma_2)&=\delta_8\{6,2,7,6,6,5,1,2,7,6,6,5,1,2,7,6,6,[5]\},\\
Y_e^{15}(\Sigma_2)&=\delta_8\{5,7,2,6,6,5,1,2,7,6,6,5,1,2,7,6,6,[5]\},\\
Y_e^{16}(\Sigma_2)&=\delta_8\{7,6,6,5,1,2,7,6,6,5,1,2,7,6,6,5,1,[2]\}.
\end{aligned} \tag{9.2.12}$$

将算法 9.1.1 用于观测数据 (9.2.11) 即可辨识出

$$L_1=\delta_{16}[12,14,4,6,3,5,11,13,11,9.3,1,4,2,12,10];$$
$$H=\delta_8[2,2,1,3,6,6,5,7,6,6,5,7,6,6,5,7].$$

同样, 由观测数据 (9.2.12) 可辨识出

$$L_2=\delta_{16}[10,16,2,8,1,7,9,15,11,9,3,1,4,2,12,10].$$

于是可得

$$L=[L_1,L_2].$$

最后, 利用 (L,H) 返回到逻辑系统可得

$$\begin{cases} X_1(t+1)=X_2(t)\bar{\vee}X_3(t),\\ X_2(t+1)=X_1(t)\to X_4(t),\\ X_3(t+1)=(\neg X_4(t)\wedge U(t))\vee\left[(X_1(t)\leftrightarrow X_4(t))\wedge\neg U(t)\right],\\ X_4(t+1)=X_2(t)\bar{\vee}X_1(t),\end{cases}\tag{9.2.13}$$
$$Y_1(t)=X_1(t)\wedge X_2(t),$$
$$Y_2(t)=X_3(t)\vee X_4(t),$$
$$Y_3(t)=\neg X_3(t).$$

9.3　混合值逻辑系统的辨识

前面讨论的布尔网络及布尔控制网络的辨识方法, 对一般逻辑网络一样可用. 例如, 设 (9.2.1) 为一混合值逻辑系统, 这里,

$$\begin{cases} X_i(t)\in\mathcal{D}_{k_i}, & i=1,2,\cdots,n,\\ U_\alpha(t)\in D_{r_\alpha}, & \alpha=1,2,\cdots,m,\\ Y_j(t)\in\mathcal{D}_{\mu_j}, & j=1,2,\cdots,p.\end{cases}$$

注 9.3.1　记 $k=\prod_{i=1}^n k_i$, $r=\prod_{\alpha=1}^m r_\alpha$, $\mu=\prod_{j=1}^p \mu_j$. 那么,

(i) 共有 r 个布尔网络子系统, 记作 Σ_s, $s=1,2,\cdots,r$;

(ii) 每个有效观测数据长度为 $k+1$, 每组相容有效观测数据集有 r 个有效观测数据;

(iii) 当每个子系统 Σ_s 均可观时, 共有 k 个不同的相容有效观测数据集.

下面用一个例子描述辨识过程.

例 9.3.1　考察如下混合值逻辑系统

$$\begin{cases} X_1(t+1)=\left[(X_1(t)\Box X_2(t))\wedge(\triangleright_3^1U(t))\right]\vee\left[(X_1(t)\circ X_2(t))\right.\\ \qquad\qquad\left.\wedge(\triangleright_3^2U(t))\right]\vee\left[(X_1(t)© X_2(t))\wedge(\triangleright_3^3U(t))\right],\\ X_2(t+1)=X_1(t)\Diamond X_2(t),\end{cases}\tag{9.3.1}$$
$$Y(t)=X_1(t)\triangle X_2(t),$$

这里, $X_1(t)\in\mathcal{D}$, $X_2(t),U(T)\in\mathcal{D}_3$, $\Box,\circ,©:\mathcal{D}\times\mathcal{D}_3\to\mathcal{D}$, $\Diamond,\triangle:\mathcal{D}\times\mathcal{D}_3\to\mathcal{D}_3$.

于是可得

(i) Σ_1:

$$\begin{cases} X_1(t+1) = X_1(t)\Box X_2(t), \\ X_2(t+1) = X_1(t)\Diamond X_2(t), \end{cases}$$
$$Y(t) = X_1(t)\triangle X_2(t). \tag{9.3.2}$$

(ii) Σ_2:

$$\begin{cases} X_1(t+1) = X_1(t)\circ X_2(t), \\ X_2(t+1) = X_1(t)\Diamond X_2(t), \end{cases}$$
$$Y(t) = X_1(t)\triangle X_2(t). \tag{9.3.3}$$

(iii) Σ_3:

$$\begin{cases} X_1(t+1) = X_1(t)\copyright X_2(t), \\ X_2(t+1) = X_1(t)\Diamond X_2(t), \end{cases}$$
$$Y(t) = X_1(t)\triangle X_2(t). \tag{9.3.4}$$

现在假定一组完备的相容有效输出序列集如下:

$$\begin{array}{ccc} Y_e^1(\Sigma_1) & Y_e^1(\Sigma_2) & Y_e^1(\Sigma_3) \\ Y_e^2(\Sigma_1) & Y_e^2(\Sigma_2) & Y_e^2(\Sigma_3) \\ \vdots & \vdots & \vdots \\ Y_e^6(\Sigma_1) & Y_e^6(\Sigma_2) & Y_e^6(\Sigma_3) \end{array} \tag{9.3.5}$$

这里

$$\begin{aligned} Y_e^1(\Sigma_1) &= \delta_3\{2,3,1,3,2,3,2,[3]\}, \\ Y_e^2(\Sigma_1) &= \delta_3\{2,3,2,3,2,3,2,[3]\}, \\ Y_e^3(\Sigma_1) &= \delta_3\{3,1,3,2,3,2,3,[2]\}, \\ Y_e^4(\Sigma_1) &= \delta_3\{1,3,2,3,2,3,2,[3]\}, \\ Y_e^5(\Sigma_1) &= \delta_3\{3,2,3,2,3,2,3,[2]\}, \\ Y_e^6(\Sigma_1) &= \delta_3\{1,1,1,1,1,1,1,[1]\}. \end{aligned} \tag{9.3.6}$$

$$
\begin{aligned}
Y_e^1(\Sigma_2) &= \delta_3\{2,1,3,2,1,3,2,[1]\},\\
Y_e^2(\Sigma_2) &= \delta_3\{2,3,3,3,3,3,3,[3]\},\\
Y_e^3(\Sigma_2) &= \delta_3\{3,2,1,3,2,1,3,[2]\},\\
Y_e^4(\Sigma_2) &= \delta_3\{1,2,3,3,3,3,3,[3]\},\\
Y_e^5(\Sigma_2) &= \delta_3\{3,3,3,3,3,3,3,[3]\},\\
Y_e^6(\Sigma_2) &= \delta_3\{1,3,2,1,3,2,1,[3]\}.
\end{aligned}
\tag{9.3.7}
$$

$$
\begin{aligned}
Y_e^1(\Sigma_3) &= \delta_3\{2,3,2,3,2,3,2,[3]\},\\
Y_e^2(\Sigma_3) &= \delta_3\{2,3,3,3,3,3,3,[3]\},\\
Y_e^3(\Sigma_3) &= \delta_3\{3,2,3,2,3,2,3,[2]\},\\
Y_e^4(\Sigma_3) &= \delta_3\{1,3,3,3,3,3,3,[3]\},\\
Y_e^5(\Sigma_3) &= \delta_3\{3,3,3,3,3,3,3,[3]\},\\
Y_e^6(\Sigma_3) &= \delta_3\{1,3,2,3,2,3,2,[3]\}.
\end{aligned}
\tag{9.3.8}
$$

利用 (9.3.6)—(9.3.8), 不难算出

$$
\begin{aligned}
L_1 &= \delta_6[3,5,4,5,2,6],\\
L_2 &= \delta_6[6,5,1,2,5,2],\\
L_3 &= \delta_6[3,5,,5,5,3],\\
H &= \delta_3[2,2,3,1,3,1].
\end{aligned}
\tag{9.3.9}
$$

最后可得, 式 (9.3.1) 中的算子的结构矩阵分别为

$$
\begin{aligned}
M_{\Box} &= \delta_2[1,2,2,2,1,2],\\
M_{\Diamond} &= \delta_3[3,2,1,2,2,3],\\
M_{\circ} &= \delta_2[2,2,1,1,2,1],\\
M_{\circledcirc} &= \delta_2[1,2,1,2,2,1],\\
M_{\triangle} &= \delta_3[2,2,3,1,3,1].
\end{aligned}
\tag{9.3.10}
$$

第 10 章 输 出 调 节

输出调节 (output regulation) 是现代控制理论中的一个重要课题. 对于线性系统, 输出调节问题最早是由 Johnson 提出的[69]. 它的解决催生了著名的内模原理 (internal model principle)[120]. 后来, 它被从线性系统推广到非线性系统, Isidori 和黄捷教授是非线性系统输出调节方面的两位著名代表人物[64].

布尔网络的输出调节问题比较简单, 它不会有轨道逃逸或其他内稳问题, 因此, 输出调节问题与跟踪问题是一样的. 这方面的研究包括: 固定输出值的跟踪[45, 91]、有限长信号跟踪[95], 以及一般信号跟踪[146], 等等. 本章的主要内容来自文献 [146].

10.1 控制不变集

给定一个布尔控制网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), U_2(t), \cdots, U_m(t)), \\ X_2(t+1) = F_2(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), U_2(t), \cdots, U_m(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F_n(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), U_2(t), \cdots, U_m(t)), \end{cases} \tag{10.1.1}$$
$$Y_j(t) = h_j(t), \quad j = 1, 2, \cdots, p,$$

这里, $X_i(t) \in \mathcal{D}$, $i = 1, 2, \cdots, n$ 为状态变量, $U_\alpha(t) \in \mathcal{D}$, $\alpha = 1, 2, \cdots, m$ 为控制变量, $Y_j(t) \in \mathcal{D}$, $j = 1, 2, \cdots, p$ 为输出 (或曰观测) 变量. $F_i : \mathcal{D}^{n+m} \to \mathcal{D}$, $h_j : \mathcal{D}^n \to \mathcal{D}$ 为布尔函数.

记 $F = (F_1, F_2, \cdots, F_n)$, 设 F 的结构矩阵为 L, $h = (h_1, h_2, \cdots, h_p)$, h 的结构矩阵为 H, 则 (10.1.1) 的代数状态空间表示为

$$\begin{aligned} x(t+1) &= Lu(t)x(t), \\ y(t) &= Hx(t), \end{aligned} \tag{10.1.2}$$

这里 $x_i(t) = \vec{X}_i(t)$, $u_\alpha(t) = \vec{U}_\alpha(t)$, $y_j(t) = \vec{Y}_j(t)$, $x(t) = \ltimes_{i=1}^n x_i(t) \in \Delta_{2^n}$, $u(t) = \ltimes_{\alpha=1}^m u_\alpha(t) \in \Delta_{2^m}$, $y(t) = \ltimes_{j=1}^p y_j(t) \in \Delta_{2^p}$.

定义 10.1.1 考虑布尔控制网络 (10.1.1). 给定一个状态子集 $V \subset \Delta_{2^n}$.

(i) V 的一个子集 $I^V \subset V$ 称为 V 的控制不变子集, 如果对每一个 $X \in I^V$ 至少存在一个控制 $U \in \Delta_{2^m}$, 使得 $Lux \in I^v$. 这里, $x = \vec{X}$, $u = \vec{U}$, $I^v = \{\vec{Z} \mid Z \in I^V\}$. 即

$$Lux \in I^v, \quad \forall x \in I^v,\ \exists u, \tag{10.1.3}$$

这里, u 依赖于 x,

(ii) I^V_m 称为 V 的最大控制不变集, 如果它是 V 的控制不变子集, 且包含所有 V 的控制不变子集. 即

$$I^V \subset I^V_m.$$

注 10.1.1 (i) 给定 $V \subset \Delta_{2^n}$, 则 I^V_m 存在且唯一. 因为根据定义, 它是所有 V 的控制不变集的并. 但它可能是空集.

(ii) 因为 Δ_{2^n} 是一个离散集, 我们可将其设为一个具有离散拓扑的拓扑空间. 因此, 所有子集均可视为其拓扑子空间. 但通常只称它为状态子集[56], 这是因为逻辑系统中是将逻辑函数集合称为子空间.

定义 10.1.2 考察布尔控制网络 (10.1.1).

(i) 矩阵

$$M := \sum_{j=1}^{2^m} \mathcal{B}L\delta_{2^m}^j \tag{10.1.4}$$

称为控制转移矩阵.

(ii) 设状态子集

$$v = \vec{V} = \left\{\delta_{2^n}^{i_1}, \delta_{2^n}^{i_2}, \cdots, \delta_{2^n}^{i_k}\right\}. \tag{10.1.5}$$

矩阵 $M_V \in \mathcal{B}_{k\times k}$, 称为 V 上的控制转移矩阵, 它由矩阵 M 删去所有的第 i 行与第 i 列生成, 这里, $i \neq i_1, i_2, \cdots, i_k$.

定义 10.1.3 设 $V \subset \mathcal{D}^n$.

(i) 一个点 $X^0 \in V$ 称为 V 的控制不动点, 如果存在控制 U 使 $F_i(U, X^0) = X^0_i$, $i = 1, 2, \cdots, n$ (或其代数状态空间表达式满足 $Lux^0 = x^0$);

(ii) 一组点集 $\{X^0, X^1, \cdots, X^\ell = X^0\} \subset V$ 称为 V 的控制极限环, 如果存在 U^i, $i = 0, 1, \cdots, \ell-1$, 使得 $F_j(U^i, X^i) = X^{i+1}_j$, $i = 0, 1, \cdots, \ell-1$, $j = 1, 2, \cdots, n$ (或其代数状态空间表达式满足 $Lu^ix^i = x^{i+1}$, $i = 0, 1, \cdots, \ell-1$), 并且, $x^0, x^1, \cdots, x^{\ell-1}$ 均不相等;

(iii) V 的所有控制不动点与控制极限环的并集称为 V 的控制吸引集.

例 10.1.1 考察一个布尔控制网络

$$\begin{cases} X_1(t+1) = \big[(X_1(t) \leftrightarrow X_2(t)) \vee U(t)\big] \vee (\neg U(t)), \\ X_2(t+1) = \big[(X_1(t) \vee (\neg X_1(t) \wedge \neg X_2(t))) \wedge U(t)\big] \big[\neg(X_1(t) \vee X_2(t)) \wedge \neg U(t)\big]. \end{cases} \tag{10.1.6}$$

不难得到其代数状态空间表达式:

$$x(t+1) = Lu(t)x(t), \tag{10.1.7}$$

这里

$$L = \delta_4[1,3,4,1,2,2,2,1].$$

其控制状态转移图见图 10.1.1 所示.

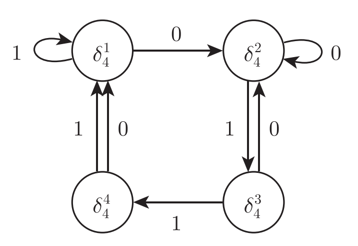


图 10.1.1 (10.1.6) 的控制状态转移图

考虑集合 V, 其元素的向量表示为

$$v = \vec{V} = \delta_4\{2,3,4\}.$$

它有一个控制不动点为 δ_4^2, 一个控制极限环为 $\delta_4^2 \leftrightarrow \delta_4^3$. 因此, V 的最大不变集为

$$I_m^v = \{\delta_4^2, \delta_4^3\}.$$

它也是 V 的吸引集.

命题 10.1.1 设

$$D = (d_{i,j}) := \sum_{t=1}^{k} \mathcal{B}\left(M_V\right)^{(t)}, \tag{10.1.8}$$

这里 $k = |V|$, 又令

$$A^V := \{\delta_{2^n}^{i_\beta} \mid d_{\beta,\beta} = 1\}. \tag{10.1.9}$$

A^V 是 V 的控制不变集.

证明 首先, $x=\delta_k^{i_\alpha}\in v:=\vec{V}$ 属于控制不变集, 当且仅当, 至少有一个 $u=\delta_{2^m}^j$ 使得 $L\delta_{2^m}^j$ 的 (i_α,i_α) 元素为 1. 因此, M_v 的 (α,α) 元素为 1. 反之, 若 M_v 的 (α,α) 元素为 1, 则至少存在一个 $u=\delta_{2^m}^j$, 使 $L\delta_{2^m}^j$ 的 (i_α,i_α) 元素为 1.

类似地, 设 $x=\delta_k^{i_\alpha}\in v$ 是长度 $s>1$ 的控制极限环的一个元素, 当且仅当, $(M_V)^{(t)}$ 的 (α,α) 元素 $(M_V)^{(t)}=0, t<s$, 并且, $(M_V)^{(s)}$ 的 (α,α) 元素 $(M_V)^{(s)}=1$.

因为 V 内最长控制极限环长度 $s\leqslant k$, 结论显见. □

定义 10.1.4 (i) 设 $A\subset\Delta_{2^n}$. A 的控制前身, 记作 $P(A)$, 定义为

$$P(A)=L^{-1}(A):=\{z\in\Delta_{2^n}\mid\exists u,\ 使得\ Luz\in A\}.$$

(ii) 设 $A\subset V\subset\Delta_{2^n}$. A 的 V 控制前身, 记作 $P_V(A)$, 定义为

$$P_V(A)=P(A)\cap V.$$

以下命题直接来自定义.

命题 10.1.2 设 V 由 (10.1.5) 确定, $A\subset V$. 定义 A 的示性矩阵如下:

$$C_A:=\sum_{\delta_{2^n}^{i_s}\in A}\mathcal{B}\delta_k^s. \tag{10.1.10}$$

那么, $\delta_{2^n}^{i_j}\in P_V(A)$, 当且仅当,

$$\mathrm{Col}_j(M_V)\wedge C_A\neq\mathbf{0}_k. \tag{10.1.11}$$

证明 考察最初的状态空间, 根据 M_V 的构造, 不难发现 (10.1.11) 表明

$$\mathrm{Col}_{i_j}(M)\wedge\bar{C}_A\neq\mathbf{0}_{2^n}, \tag{10.1.12}$$

这里

$$\bar{C}_A:=\sum_{\delta_{2^n}^t\in A}\mathcal{B}\delta_{2^n}^t.$$

注意到

$$\mathrm{Col}_{i_j}(M)=M\delta_{2^n}^{i_j}.$$

因此, 等式 (10.1.12) 说明, 至少有一个 u_0, 使得

$$Lu_0\delta_{2^n}^{i_j}\in A.$$

即 $\delta_{2^n}^{i_j}\in P(A)$.

反之, 如果 (10.1.11) 不成立, 即

$$\mathrm{Col}_j(M_V) \wedge C_A = \mathbf{0}_k.$$

类似的讨论表明: 若

$$\mathrm{Col}_{i_j}(M) \wedge \bar{C}_A = \mathbf{0}_{2^n},$$

则 $\delta_{2^n}^{i_j} \notin A$;

若

$$\mathrm{Col}_{i_j}(M) \wedge \bar{C}_A \neq \mathbf{0}_{2^n},$$

则 $\delta_{2^n}^{i_j} \notin V$.

但是

$$\left\{\mathrm{Col}_{i_j}(M) \wedge \bar{C}_A\right\}\Big|_V = \mathbf{0}_k.$$

因此, $\delta_{2^n}^{i_j} \notin P_V(A)$. □

需要说明的是, 如果 $A, B \in \mathcal{B}_{m\times n}$, 即 A 与 B 为同维布尔矩阵, 记

$$A\sigma B := (a_{i,j}\sigma b_{i,j}),$$

这里 σ 为一个二元算子, 如 $\wedge$, $\vee$ 等. 这个记号前面已用到.

利用公式 (10.1.11) 及等式 (10.1.12), 容易计算 $P_V(A)$.

下面给出 $V \subset \Delta_{2^n}$ 的最大控制不变集的算法.

算法 10.1.1 设 $V \subset \Delta_{2^n}$.

第一步: 计算 M_V.

第二步: 利用 (10.1.5)—(10.1.9) 计算 V 的最大控制不变集 A^V.

第三步: 设 $B_0 := A^V$, 利用 (10.1.11) 迭代计算

$$B_s := P_V(B_{s-1}) \cap B_{s-1}^c, \quad s = 1, 2, \cdots,$$

这里, B_s^c 是 B_s 的余集, 即, $B_s^c = \Delta_{2^n} \backslash B_s$.

第四步: 如果 $B_s = \varnothing$, 则令

$$\Psi = \bigcup_{j=0}^{s-1} B_j. \tag{10.1.13}$$

终止算法.

命题 10.1.3 利用算法 10.1.1, 由 (10.1.13) 生成的 Ψ 是 V 的最大控制不变集, 即

$$I_m^V = \Psi. \tag{10.1.14}$$

证明　设 $x_0 \in I_m^V$. 令 $x(0)=x_0$, 根据定义, 则存在 u_i, $i=0,1,\cdots$, 使得

$$x(t+1)=Lu(t)x(t)\in V, \quad t=0,1,2,\cdots.$$

因为 V 是有限集, 经过有限步必有 $x(T)\in\{x(0),x(1),\cdots,x(T-1)\}$. 于是, 在 T 步后从 $x(0)$ 出发的轨线必定进入 A^V. 因此, 经 $T\geqslant 0$ 步后, $x(0)\in B_T$. 于是 $x_0\in\Psi$. 因此, $I_m^V\subset\Psi$.

另一方面, 依据构造可知 Ψ 是 V 的控制不变子集. 因此, $\Psi\subset I_m^V$. 于是有

$$\Psi = I_m^V.$$

□

下面考虑一个例子.

例 10.1.2　考察如下布尔控制网络:

$$\begin{cases} X_1(t+1)= & U(t)\wedge\{X_1(t)\vee[\neg X_1(t)\wedge(\neg X_2(t)\wedge X_3(t))]\} \\ & \vee\neg U(t)\wedge\{X_1(t)\vee(\neg X_1(t)\wedge X_3(t))\}, \\ X_2(t+1)= & U(t)\wedge\{[X_1(t)\wedge(X_2(t)\wedge X_3(t))] \\ & \vee[\neg X_1(t)\wedge\neg X_2(t)\wedge X_3(t)]\} \\ & \vee\neg U(t)\wedge\{\neg X_1(t)\wedge X_2(t)\wedge X_3(t)\}, \\ X_3(t+1)= & U(t)\wedge\{[X_1(t)\wedge\neg(X_2(t)\vee X_3(t))] \\ & \vee[\neg X_1(t)\wedge((X_2(t)\wedge X_3(t))\vee\neg X_2(t))]\} \\ & \vee\neg U(t)\wedge\{[X_1(t)\wedge(X_2(t)\vee(\neg X_2(t)\wedge\neg X_3(t)))] \\ & \vee[\neg X_1(t)\wedge(\neg(X_2(t)\wedge X_3(t)))]\}. \end{cases} \tag{10.1.15}$$

它们的分量代数状态空间表达式为

$$\begin{cases} x_1(t+1)=M_1u(t)x(t), \\ x_2(t+1)=M_2u(t)x(t), \\ x_3(t+1)=M_3u(t)x(t), \end{cases}$$

这里, $x(t)=x_1(t)x_2(t)x_3(t)$, 且

$$\begin{aligned} M_1&=\delta_2[1,1,1,1,2,2,1,2,1,1,1,1,1,2,1,2], \\ M_2&=\delta_2[1,2,2,2,2,2,1,2,2,2,2,2,1,2,2,2], \\ M_3&=\delta_2[2,2,2,1,1,2,1,1,1,1,2,1,2,1,1,1]. \end{aligned}$$

于是, 网络 (10.1.15) 的代数状态空间表达式为

$$x(t+1)=Lu(t)x(t), \tag{10.1.16}$$

这里

$$\begin{aligned} L &= M_1 * M_2 * M_3 \\ &= \delta_8[2,4,4,3,7,8,1,7,3,3,4,3,2,7,3,7]. \end{aligned}$$

利用 (10.1.4), 网络 (10.1.15) 的控制转移矩阵为

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

现在假定子集 V 为

$$v = \vec{V} = \{\delta_8^1, \delta_8^2, \delta_8^3, \delta_8^4, \delta_8^6, \delta_8^8\}.$$

则受限控制转移阵为

$$M_V = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

据此, 并利用 (10.1.8) 可算得

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

因此, 我们有 V 的控制吸引集:

$$B_0 := A^V = \{\delta_8^3, \delta_8^4\}.$$

根据 (10.1.10), 可得

$$C_A = [0,0,1,1,0,0]^{\mathrm{T}}.$$

再由命题 10.1.2 及等式 (10.1.11) 不难算出

$$\begin{aligned} B_1 &= P_V(B_0) \cap B_0^c = \{\delta_8^1, \delta_8^2\}, \\ B_2 &= P_V(B_1) \cap B_1^c = \varnothing. \end{aligned}$$

最后可得, V 的最大控制不变集为

$$I_m^V = \{\delta_8^1, \delta_8^2, \delta_8^3, \delta_8^4\} \subset V. \tag{10.1.17}$$

10.2　输出调节的集合能控性方法

10.2.1　参考系统轨道跟踪

给定参考系统

$$\begin{cases} Z_1(t+1) = R_1(Z_1(t), \cdots, Z_p(t)), \\ Z_2(t+1) = R_2(Z_1(t), \cdots, Z_p(t)), \\ \qquad\qquad \cdots\cdots \\ Z_p(t+1) = R_p(Z_1(t), \cdots, Z_p(t)), \end{cases} \tag{10.2.1}$$

其代数状态空间表达式为

$$z(t+1) = Gz(t), \tag{10.2.2}$$

这里, $z(t) = \ltimes_{i=1}^p z_i(t)$, $G \in \mathcal{L}_{2^p \times 2^p}$.

定义 10.2.1　考察布尔控制网络 (10.1.1) 及参考网络 (10.2.1). 称轨道跟踪问题可解, 如果对任何初始状态 $X(0) = X_0$ 及 $Z(0) = Z_0$, 存在有限时间 $T \geqslant 0$ (它可能依赖于 X_0 及 Z_0), 以及一个控制序列 $U(t)$, $t \geqslant 0$, 使得

$$Y(t) = Z(t), \quad t \geqslant T. \tag{10.2.3}$$

为利用集合能控性, 构造辅助系统

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ z(t+1) = Gz(t). \end{cases} \tag{10.2.4}$$

设 $w(t)=x(t)z(t)$, 可得

$$\begin{aligned}w(t+1)&=Lu(t)x(t)Gz(t)\\&=L\left(I_{2^{n+m}}\otimes G\right)u(t)w(t).\end{aligned}$$

于是可得

$$w(t+1)=Fu(t)w(t), \tag{10.2.5}$$

这里, $F=L\left(I_{2^{n+m}}\otimes G\right)$.

构造 (10.2.5) 的能控性矩阵

$$\mathcal{C}=\sum_{j=1}^{2^{n+p}}\mathcal{B}M^{(j)}, \tag{10.2.6}$$

这里

$$M=\sum_{i=1}^{2^m}\mathcal{B}F\delta_{2^m}^i. \tag{10.2.7}$$

基于 (10.2.5), 定义集合

$$V:=\{(x,z)\mid Hx=z\}. \tag{10.2.8}$$

利用算法 10.1.1, 可计算辅助系统 (10.2.4) 关于 V 的最大控制不变集 I_m^V. 然后, 关于轨道跟踪可得以下结论.

命题 10.2.1　*考察布尔控制网络 (10.1.1) 及参考网络 (10.2.1). 参考网络轨道跟踪问题可解, 当且仅当, 对任何初值 $X(0)$ 及 $Z(0)$, 构造辅助系统的初值 $w(0)=\vec{W}(0)=x(0)z(0)$, 则存在一个 $T\geqslant 0$ 和一族控制序列 $\{U(0),U(1),\cdots,U(T-1)\}$, 使相应轨线 $w(t)=(x(t)z(t))$ 满足 $W(T)\in I_m^V$, 这里, V 满足 (10.2.8).*

证明　(充分性) 如果存在控制序列 $\{U(0),U(1),\cdots,U(T-1)\}$, 使 $W(T)\in I_m^V$, 那么, 可知 $y(T+1)=Lx(T+1)=z(T+1)$. 同时, 因为 I_m^V 是控制不变集, 一旦轨线进入该集, 则存在控制序列, 使轨线永远保持在 I_m^V 里边.

(必要性) 假定存在一个点 $w(0)=x(0)z(0)$, 不存在一个控制序列能将其导入最大控制不变集 I_m^V. 但却存在一个控制序列 $\{U(t)\mid t=0,1,\cdots\}$, 它能解决轨道跟踪问题. 即, 存在 $T>0$ 使得 $Y(t)=Z(t)$, $t>T$. 那么, 我们有 $W(t)\in V$, 当 $t>T$. 因为 V 是个有限集, 正如我们在证明命题 10.1.3 时指出的, $W(t)$ 一定要进入某个 V 中的不变集 A^V. 矛盾. □

设

$$P^0=\{\{\delta_{2^n}^1,\delta_{2^p}^1\},\{\delta_{2^n}^1,\delta_{2^p}^2\},\cdots,\{\delta_{2^n}^1,\delta_{2^p}^{2^p}\},\cdots,\{\delta_{2^n}^{2^n},\delta_{2^p}^1\},\cdots,\{\delta_{2^n}^{2^n},\delta_{2^p}^{2^p}\}\}\sim\varDelta_{2^{n+p}},$$

以及

$$P^d = \{E\},$$

这里 $E := I_m^V$, 并设 V 由 (10.2.8) 决定.

注意到 $\Delta_{2^{n+p}}$ 中的每个点都属于 P^0, 而 P^d 只有一个元素, 即集合 E. 也就是说, 目标集由状态空间中所有满足 (10.2.8) 的点组成. 因此, 初始点集合的示性矩阵为 $J_0 = I_{2^{n+p}}$, 目标点集合的示性矩阵为 $J_d = J_{\{E\}}$. 因此, 集合能控性矩阵为

$$\mathcal{C}_S = J_d^{\mathrm{T}} \times_{\mathcal{B}} \mathcal{C} \times_{\mathcal{B}} J_0 = J_{\{E\}}^{\mathrm{T}} \times_{\mathcal{B}} \mathcal{C}. \tag{10.2.9}$$

根据以上的讨论并利用命题 10.2.1, 可以得到如下结果.

定理 10.2.1　考察布尔控制网络 (10.1.1) 及参考网络 (10.2.1). 参考系统轨道跟踪问题可解, 当且仅当, 由 (10.2.9) 定义的集合能控性矩阵为正矩阵, 即

$$\mathcal{C}_S > 0. \tag{10.2.10}$$

10.2.2　参考系统的输出跟踪

轨道跟踪的设计方法也可以用到参考系统的输出跟踪上.

考察一个参考系统

$$\begin{cases} \hat{Z}_1(t+1) = \hat{g}_1(\hat{Z}_1(t), \cdots, \hat{Z}_q(t)), \\ \hat{Z}_2(t+1) = \hat{g}_2(\hat{Z}_1(t), \cdots, \hat{Z}_q(t)), \\ \qquad\qquad \cdots\cdots \\ \hat{Z}_q(t+1) = \hat{g}_q(\hat{Z}_1(t), \cdots, \hat{Z}_q(t)), \end{cases}$$
$$\hat{Y}(t) = \psi_j(\hat{Z}_1(t), \cdots, \hat{Z}_q(t)), \quad j = 1, \cdots, p. \tag{10.2.11}$$

其代数状态空间表达式为

$$\begin{cases} \hat{z}(t+1) = \hat{G}\hat{z}(t), \\ \hat{y}(t) = \Psi \hat{z}(t), \end{cases} \tag{10.2.12}$$

这里, $\hat{z}_i = \vec{\hat{Z}}_i$, $i = 1, 2, \cdots, q$, $\hat{y}_j = \vec{\hat{Y}}_j$, $j = 1, 2, \cdots, p$, $\hat{z}(t) = \ltimes_{i=1}^q \hat{z}_i(t)$, $\hat{G} \in \mathcal{L}_{2^q \times 2^q}$, $\hat{y}(t) = \ltimes_{j=1}^p \hat{y}_i(t)$, $\Psi \in \mathcal{L}_{2^p \times 2^q}$.

定义 10.2.2　考察布尔控制网络 (10.1.1) 及参考网络 (10.2.11). 称参考网络输出跟踪问题可解, 如果对任何初值 $X(0) = X_0$ 及 $\hat{Z}(0) = \hat{Z}_0$, 存在有限时间 $T \geqslant 0$ 和一个控制序列 $U(t)$, $t \geqslant 0$, 使得

$$Y(t) = \hat{Y}(t), \quad t \geqslant T. \tag{10.2.13}$$

给定布尔控制网络 (10.1.1) 及参考网络 (10.2.11). 我们可构造一个辅助系统如下:

$$\begin{cases} x(t+1) = Lu(t)x(t), \\ \hat{z}(t+1) = \hat{G}\hat{z}(t). \end{cases} \tag{10.2.14}$$

类似于参考系统轨道跟踪问题, 定义目标集合 $\hat{V}$ 如下:

$$\hat{V} := \{(x, \hat{z}) \mid Hx = \Psi\hat{z}\}. \tag{10.2.15}$$

再构造

$$\hat{E} := I_m^{\hat{V}}. \tag{10.2.16}$$

利用集合能控性, 参考系统的轨道跟踪与输出跟踪的主要区别就在于构造相应的最大控制不变集 E 和 $\hat{E}$. 显然, 只要将集合 E 用 $\hat{E}$ 代替, 将参考系统 (10.2.1) 用参考系统 (10.2.11) 代替, 则命题 10.2.1 和定理 10.2.1 依然成立. 我们将此写成定理 10.2.1 的一个推论.

推论 10.2.1　给定布尔控制网络 (10.1.1) 及参考网络 (10.2.11). 参考网络输出跟踪问题可解, 当且仅当, 由

$$\hat{\mathcal{C}}_S := J_{\{\hat{E}\}}^{\mathrm{T}} \times_{\mathcal{B}} \mathcal{C} \tag{10.2.17}$$

定义的集合能控性矩阵为正矩阵.

在结束本节前, 我们给出一个算法改进. 它在理论上是很直观的, 但它可减小验证可解性时的计算复杂性.

命题 10.2.2　在定理 10.2.1 或推论 10.2.1 中, $E = I_m^V$ 和 $\hat{E} = I_m^{\hat{V}}$ 如果用 $E = A^V$ 以及 $\hat{E} = A^{\hat{V}}$ 来代替, 即用吸引集代替最大控制不变集, 那么, 定理 10.2.1 或推论 10.2.1 的结论仍然成立.

证明　充分性是显然的. 至于必要性, 注意到如果一条轨道从 w_0 (或 $\hat{w}_0 = x_0\hat{z}_0$) 到最大控制不变集, 它必然要到达吸引集. 不然的话, 这就意味着轨线要在集合 $I_m^V \setminus A^V$ 上跳动. 那么, 根据定义 10.1.3, 必然存在另一个控制吸引集 $\tilde{A}^V \subset I_m^V \setminus A^V$, 并且, 根据命题 10.1.1, $\tilde{A}^V = A^V$, 这导致矛盾. □

命题 10.2.2 的好处在于: 为检验跟踪问题的可解性, 我们无须找出最大控制不变集, 而只要找出吸引集就可以了. 尽管此后, 如果要设计最佳控制, 还是要计算最大控制不变集的.

10.3 输出调节控制设计

10.3.1 开环控制设计

10.3.1.1 轨道跟踪

设 V 中的最大不变集为 $E=I_m^V=\{\delta_{2^{n+p}}^{j_1},\cdots,\delta_{2^{n+p}}^{j_s}\}$, 这里, $\delta_{2^{n+p}}^{j_\ell}=\delta_{2^n}^{i_\ell}\ltimes\delta_{2^p}^{\lambda_\ell}$. 那么, $j_\ell=(i_\ell-1)2^p+\lambda_\ell$, $\ell=1,2,\cdots,s$.

下面给出一个算法, 它能构造一个控制序列, 该序列是布尔控制网络 (10.1.1) 的输出跟踪 (10.2.1) 的轨线. 并且无论初值 $(x(0),z(0))$ 是多少, 它均能保证最短时间跟上.

算法 10.3.1 设 $x(0)=x_0=\delta_{2^n}^i$, $z(0)=z_0=\delta_{2^p}^j$, 且 $x_0\ltimes z_0=\delta_{2^{n+p}}^l$.

第一步: 如果 $(x_0,z_0)\notin E$, 找出最小的 k, 使 $\mathrm{Col}_l(\mathcal{C}_S^k)=1$, 这里

$$\mathcal{C}_S^k=J_{\{E\}}^{\mathrm{T}}\times_{\mathcal{B}}\sum_{j=1}^k\mathcal{B}M^{(j)}.$$

计算 $z(k)=G^kz(0)=\delta_{2^p}^{\lambda_k}$, 从而有

$$Hx(k)=HLu(k-1)x(k-1)=z(k)=\delta_{2^p}^{\lambda_k}.$$

利用反推的设计方法[136], 可以找到控制序列 $\{u(0),u(1),\cdots,u(k-1)\}$, 使得对任何 $x_0=\delta_{2^n}^i$ 以及 $z_0=\delta_{2^p}^j$ 均有 $y(k)=z(k)$, $k\geqslant 1$. 否则, 进入下一步.

第二步: 设 $(x(k),z(k))\in E$, $k\geqslant 0$, 此时, 需要找出一步控制 $u(k)$, 使得 $(x(k+1),z(k+1))\in E$. 设 $x(k)=\delta_{2^n}^{i_k}$ 且 $z(k)=\delta_{2^p}^{\lambda_k}$, 那么, $z(k+1)=Gz(k):=\delta_{2^p}^{\lambda_{k+1}}$. 于是可以任选一个 α, 使得 $[HL\delta_{2^m}^\alpha]_{\lambda_{k+1},i_k}=1$. 令 $u(k):=\delta_{2^m}^\alpha$.

第三步: 置 $k:=k+1$, 返回第二步.

10.3.1.2 输出跟踪

令 $\hat{E}=I_m^{\hat{V}}=\{\delta_{2^{n+q}}^{j_1},\cdots,\delta_{2^{n+q}}^{j_s}\}$, 这里 $\delta_{2^{n+q}}^{j_\ell}=\delta_{2^n}^{i_\ell}\ltimes\delta_{2^q}^{\lambda_\ell}$, 于是 $j_\ell=(i_\ell-1)2^q+\lambda_\ell$, $\ell=1,2,\cdots,s$. 以下的算法可用于构造一个控制序列, 它可令布尔控制网络 (10.1.1) 的输出在最短时间内跟上参考网络 (10.2.11) 的输出, 无论初态 $(x(0),\hat{z}(0))$ 为何值.

算法 10.3.2 设 $x(0)=x_0=\delta_{2^n}^i$, $\hat{z}(0)=\hat{z}_0=\delta_{2^q}^j$, 且 $x_0\ltimes\hat{z}_0=\delta_{2^{n+q}}^l$.

第一步: 如果 $(x_0,\hat{z}_0)\notin\hat{E}$, 找出最小的 k, 使得 $\mathrm{Col}_l(\mathcal{C}_S^k)=1$, 这里

$$\mathcal{C}_S^k=J_{\{\hat{E}\}}^{\mathrm{T}}\times_{\mathcal{B}}\sum_{j=1}^k\mathcal{B}M^{(j)}.$$

计算 $\hat{y}(k)=\Psi\hat{G}^k\hat{z}(0)=\delta_{2^p}^{\xi_k}$, 则可得

$$Hx(k)=HLu(k-1)x(k-1)=\hat{y}(k)=\delta_{2^p}^{\xi_k}.$$

利用反推的设计方法可以找到控制序列 $\{u(0),u(1),\cdots,u(k-1)\}$, 使对任何初始值 $x_0=\delta_{2^n}^i$ 及 $\hat{z}_0=\delta_{2^q}^j$, 均满足 $y(k)=\hat{y}(k)$, $k\geqslant 1$, 否则, 进入下一步.

第二步: 设 $(x(k),\hat{z}(k))\in\hat{E}$, $k\geqslant 0$, 此时, 需要找出一步控制 $u(k)$, 使得 $(x(k+1),\hat{z}(k+1))\in\hat{E}$. 设 $x(k)=\delta_{2^n}^{i_k}$, 计算 $\hat{y}(k+1)=\Psi\hat{G}\hat{z}(k):=\delta_{2^p}^{\xi_{k+1}}$. 于是可以任选一个 α, 使得 $[HL\delta_{2^m}^{\alpha}]_{\xi_{k+1},i_k}=1$. 令 $u(k):=\delta_{2^m}^{\alpha}$.

第三步: 置 $k:=k+1$, 返回第二步.

10.3.2 闭环控制设计

通常闭环控制具有更好的鲁棒性, 因此我们也希望找到输出调节的闭环解. 设闭环控制为

$$u(t)=Kx(t)z(t).$$

从初始点 $(X(0),Z(0))\sim\delta_{2^{n+p}}^i$ 出发, 记 $t=k_i$ 为最短的时间, 使得存在 $\{U_0,U_1,\cdots,U_{t-1}\}$, 该控制将 $(X(0),Z(0))$ 导入 $(X(t),Z(t))\in E$. 那么, $\{U_0,U_1,\cdots,U_{t-1}\}$ 称为初值 $(X(0),Z(0))$ 的最优控制序列.

定义 10.3.1 设 $\{U_0,U_1,\cdots,U_{t-1}\}$ 为 $(X(0),Z(0))$ 的最优控制序列, 那么, U_0 称为 $(X(0),Z(0))$ 的最优状态反馈控制, 记作 $U_0\in SF(X(0),Z(0))$, 这里 $SF(X(0),Z(0))$ 称为 $(X(0),Z(0))$ 的最优状态反馈控制集合.

下面这个结果来自动态规划的最优性原理 (optimality principle)[14].

命题 10.3.1 如果 $\{U_0,U_1,\cdots,U_{t-1}\}$ 是状态 $(X(0),Z(0))$ 的最优控制序列, 那么 U_i 就是 $(X(i),Z(i))$ 的最优状态反馈控制, 即, $U_i\in SF(X(i),Z(i))$ $(i=0,1,\cdots,t-1)$.

令 $\mathcal{O}_a:=\{\delta_{2^{n+p}}^i|k_i=a\}$, $\mathcal{O}_0:=E$, 且记 $\tilde{M}:=\sum_{j=1}^{2^m}F\delta_{2^m}^j$, 则有如下命题.

命题 10.3.2 给定初值 $(X(0),Z(0))$, 记 $x(0)=\delta_{2^n}^{\theta}$, $z(0)=\delta_{2^p}^{\lambda}$ 且 $x(0)\ltimes z(0)=\delta_{2^{n+p}}^i$, 那么,

$$|SF(x(0),z(0))|=J_{\{\mathcal{O}_{k_i-1}\}}^{\mathrm{T}}\tilde{M}\delta_{2^{n+p}}^i, \tag{10.3.1}$$

这里 $z=\delta_{2^p}^{\lambda'}:=G\delta_{2^p}^{\lambda}$.

证明 根据 $\mathcal{O}_a,a=1,2,\cdots$ 的定义可知, $(X,Z)\sim\delta_{2^{n+p}}^i\in\mathcal{O}_a\Leftrightarrow k_i=a$. 根据动态规划的最优性原理可知, 对任意 $U_0\in SF(X(0),Z(0))$ 可推出 $(X(1),Z(1))\in\mathcal{O}_{k_i-1}$. 用普通积代替布尔积, 则 (10.3.1) 可由 (10.2.9) 构造出来. 因此, (10.3.1) 表示了可以将 $(X(0),Z(0))$ 导引到最优轨线的下一步的不同控制的个数. □

根据命题 10.3.1 可知, 共有 $\prod_{j=1}^{2^{n+p}}|SF(\delta_{2^{n+p}}^j)|$ 个不同的状态反馈增益矩阵 K. 下面我们寻找从任意 $(X(0),Z(0))$ 出发, 所有的最优控制序列.

命题 10.3.3 考虑轨道跟踪问题. 设 $(x(0),z(0))\in\mathcal{O}_{k_i}$, 记 $x(0)=\delta_{2^n}^{\theta}$, $z(0)=\delta_{2^p}^{\lambda}$ 以及 $x(0)\ltimes z(0)=\delta_{2^{n+p}}^i$, 则有 $J_{\{E\}}^{\mathrm{T}}\sum_{j=1}^{k_i}M^j\delta_{2^{n+p}}^i$ 个不同的最优控制序列.

记最优控制序列为 $u_t, t=0,1,\cdots,k_i-1$, 那么, 它们可构造如下:

$$u_t\in SF(x(t),z(t))=\{\delta_{2^m}^{\alpha}\,|\,[HL\delta_{2^m}^{\alpha}]_{\lambda_t,i}=1\},\tag{10.3.2}$$

这里, $\delta_{2^p}^{\lambda_t}:=G^tz(0)=G^t\delta_{2^p}^{\lambda}$.

用 $\hat{E}$ 代替 E 并调整 X 与 Z 的维数, 那么, 定义 10.3.1、命题 10.3.1 及命题 10.3.3 对输出跟踪同样可用. 不妨将其写成一个新的命题.

命题 10.3.4 考虑输出跟踪问题. 设 $(x(0),\hat{z}(0))\in\mathcal{O}_{k_i}$, 记 $x(0)=\delta_{2^n}^{\theta}$, $\hat{z}(0)=\delta_{2^q}^{\lambda}$ 及 $x(0)\ltimes\hat{z}(0)=\delta_{2^{n+q}}^i$, 则有 $J_{\{\hat{E}\}}^{\mathrm{T}}\sum_{j=1}^{k_i}M^j\delta_{2^{n+q}}^i$ 个不同的最优控制序列.

记最优控制序列为 $u_t, t=0,1,\cdots,k_i-1$, 那么, 它们可构造如下:

$$u_t\in SF(x(t),\hat{z}(t))=\{\delta_{2^m}^{\alpha}\,|\,\left[HL\delta_{2^m}^{\alpha_j}\right]_{\xi_t,i}=1\}.\tag{10.3.3}$$

综合以上关于开环与闭环输出调节控制的结果, 不难得出如下结论.

推论 10.3.1 如果一个输出调节问题由开环控制可解, 它也必可由闭环控制解决. 因此, 对于输出调节问题, 开环控制与闭环控制等价.

10.3.3 一个生物网络的例子

例 10.3.1[111] 考察一个乳糖操控网络 (lac operon regulatory network)

$$\begin{cases}X_1(t+1)=\neg U_1(t)\wedge(X_3(t)\vee U_2(t)),\\ X_2(t+1)=X_1(t),\\ X_3(t+1)=\neg U_1(t)\wedge((X_2(t)\wedge U_2(t))\vee(X_3(t)\wedge\neg X_2(t))),\end{cases}\tag{10.3.4}$$
$$Y_1(t)=X_2(t)\bar{\vee}X_3(t),\ Y_2(t)=X_3(t),$$

这里 $X_1,X_2,X_3,U_1,U_2\in\mathcal{D}$ 分别代表核糖核酸 (mRNA)、乳糖 Z 多肽 (Lac Z polypeptide)、内乳糖 (internal lactose)、外乳糖 (external lactose) 和外葡萄糖 (external glucose). 变量取 1 表示相应分子处于高浓度, 变量取 0 表示相应分子处于低浓度.

问题一: 轨道跟踪问题.

参考系统刻画乳糖在乳糖酶作用下的酶催水解:

$$\begin{cases} Z_1(t+1) = \neg Z_1(t), \\ Z_2(t+1) = Z_1(t), \end{cases} \tag{10.3.5}$$

这里 $Z_1, Z_2 \in \mathcal{D}$ 分别代表乳糖和葡萄糖.

原系统与参考系统的代数状态空间表达式分别为

$$\begin{aligned} & x(t+1) = Lu(t)x(t), \\ & y(t) = Hx(t), \end{aligned} \tag{10.3.6}$$

这里,

$$\begin{aligned} & L = \delta_8[6,6,6,6,8,8,8,8,6,6,6,6,8,8,8,8,1,1,1,1,3,3,3,3,1,6,1,5,3,8,3,7], \\ & H = \delta_4[3,4,3,2,3,4,3,2], \end{aligned}$$

以及

$$z(t+1) = Gz(t), \tag{10.3.7}$$

这里,

$$G = \delta_4[3,3,2,2].$$

设 $w(t) = x(t)z(t)$ 并利用 (10.3.3) 及 (10.3.7), 可构造辅助系统如下:

$$w(t+1) = Fu(t)w(t), \tag{10.3.8}$$

这里,

$$F = L(I_{32} \otimes G) = \delta_{32}[23,23,22,22,\cdots,27,27,26,26] \in \mathcal{L}_{32\times 128}. \tag{10.3.9}$$

于是可构造

$$M := \sum_{i=1}^{4} \mathcal{B}F\delta_4^i = \delta_{32}[3+23,3+23,\cdots,10+26+30] \in \mathcal{B}_{32\times 32}.$$

根据 (10.2.8) 可构造 V, 然后可得 V 中的吸引集 A^V:

$$E = A^V = \delta_{32}\{27,30\}.$$

从而有

$$J_{\{E\}} = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0]^{\mathrm{T}}.$$

据此可算得

$$\mathcal{C}_S = J_{\{E\}}^{\mathrm{T}} \times_{\mathcal{B}} \sum_{i=1}^{32} \mathcal{B}M^{(i)} = \mathbf{1}_{32}^{\mathrm{T}} > 0.$$

根据定理 10.2.1 可知, 该轨道跟踪问题可解.

下面我们计算 V 中的最大不变集 I_m^V:

$$\bar{E} := I_m^V = \delta_{32}\{14, 19, 24, 27, 30\}.$$

于是有

$$J_{\{\bar{E}\}} = [0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,0,1,0,0]^{\mathrm{T}}.$$

为了找到闭环控制 $u(t) = Kx(t)z(t)$, 我们要找到所有的最优反馈增益矩阵 $K \in \mathcal{L}_{4\times 32}$.

首先, 将状态按进入 $\bar{E}$ 的最短时间分类:

$$\begin{aligned}
&\mathcal{O}_0 = \bar{E} = \delta_{32}\{14, 19, 24, 27, 30\},\\
&\mathcal{O}_1 = \delta_{32}\{13, 14, 19, 20, 23, 24, 27, 28, 29, 30, 31, 32\},\\
&\mathcal{O}_2 = \delta_{32}\{1, 2, 5, 6, 9, 10, 17, 18, 21, 22, 25, 26\},\\
&\mathcal{O}_3 = \delta_{32}\{3, 4, 7, 8, 11, 12, 15, 16\}.
\end{aligned}$$

其次, 将 $\delta_{32}^i, i = 1, 2, \cdots, 32$ 代入 (10.3.2) 以取代 $(x(t), z(t))$, 即得 $SF(\delta_{32}^i)$. 然后 $\mathrm{Col}_i(K)$ 可由 $SF(\delta_{32}^i)$ 选得:

$$\mathrm{Col}_i(K) \in \begin{cases}
\delta_4\{1, 2, 3, 4\}, & i = 3, 4, 7, 8, 11, 12, 15, 16,\\
\delta_4\{1, 2, 4\}, & i = 5, 6, 21, 22, 23, 24,\\
\delta_4\{4\}, & i = 13, 14, 29, 30,\\
\delta_4\{1, 2\}, & \text{其他}.
\end{cases}$$

由以上的讨论可知, 这里有 $2^{30} \times 3^6$ 个不同的反馈增益矩阵, 我们可以根据系统的其他要求来选择它们. 如果希望多用外乳糖和葡萄糖, 则可选

$$K = \delta_4[1,1,1,1,1,1,1,1,1,1,1,1,4,4,1,1,1,1,1,1,1,1,1,1,1,1,1,1,4,4,1,1].$$

在图 10.3.1 中, 设 $x(0)z(0) = \delta_{32}^4$ 为初始状态. 利用控制 $u(t) = Kx(t)z(t)$, 令 $T = 2$, 则得 $y(t) = z(t)$, $t \geqslant T$.

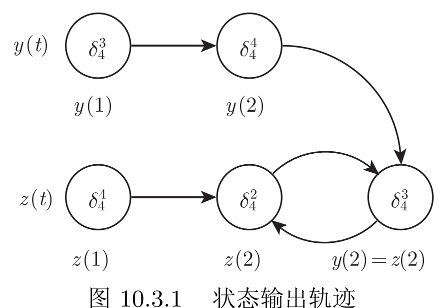


图 10.3.1　状态输出轨迹

问题二: 输出跟踪问题.

考察如下五结点系统, 它最早由文献 [95] 提出:

$$\begin{cases} \hat{Z}_1(t+1) = \hat{Z}_3(t), \\ \hat{Z}_2(t+1) = \hat{Z}_1(t), \\ \hat{Z}_3(t+1) = \hat{Z}_3(t) \vee (\hat{Z}_4(t) \wedge \hat{Z}_2(t)), \\ \hat{Z}_4(t+1) = \hat{Z}_5(t) \vee (\hat{Z}_4(t) \wedge \neg\hat{Z}_2(t)), \\ \hat{Z}_5(t+1) = \hat{Z}_1(t), \end{cases} \\ \hat{Y}(t) = \hat{Z}_1(t), \tag{10.3.10}$$

这里, $\hat{Z}_1, \hat{Z}_2, \hat{Z}_3, \hat{Z}_4, \hat{Z}_5 \in \mathcal{D}$, 分别为核糖核酸 (mRNA)、$\beta$ 半乳糖苷酶 (β galactosidase)、异乳糖 (allolactose)、胞内乳糖 (intracellular lactose) 和紫胶渗透酶 (lac permease). $\hat{Y} \in \mathcal{D}$ 为输出. 系统 (10.3.4) 中的 $Y_1(t)$ 要求跟踪系统 (10.3.10) 的输出 $\hat{Y}(t)$.

(10.3.10) 的代数状态空间表示为

$$\begin{cases} \hat{z}(t+1) = \hat{G}\hat{z}(t), \\ \hat{y} = \Psi\hat{z}(t), \end{cases} \tag{10.3.11}$$

这里,

$$\hat{G} = \delta_{32}[1,3,1,3,17,19,\cdots,30,30,30,32] \in \mathcal{L}_{32\times 32},$$

$$\Psi = \delta_2[1,1,1,1,\cdots,2,2,2,2] \in \mathcal{L}_{2\times 32}.$$

令 $w(t) = x(t)\hat{z}(t)$, 则可构造辅助系统如下:

$$w(t+1) = Fu(t)w(t), \tag{10.3.12}$$

这里,

$$F = L(I_{32} \otimes \hat{G}) = \delta_{256}[161,163,\cdots,222,224] \in \mathcal{B}_{256\times 1024}. \tag{10.3.13}$$

于是可知

$$M := \sum_{i=1}^{4} \mathcal{B}F\delta_4^i = \delta_{256}[1+161, 3+163, \cdots, 96+224+256].$$

不难算得

$$\begin{aligned}\hat{e} = \vec{\hat{E}} = I_m^{\hat{v}} = \delta_{256}\{&21-24, 29-32, 53-56, 61-64,\\ &85-88, 93-96, 101-104, 109-112,\\ &145-148, 151-160, 177-180,\\ &183-192, 209-212, 215-240\}.\end{aligned}$$

因此, 有

$$J_{\{\hat{E}\}} = \delta_{256}[21+22+\cdots+240].$$

利用这些, 即可得到

$$\mathcal{C}_S = J_{\{\hat{E}\}}^{\mathrm{T}} \sum_{i=1}^{256} \mathcal{B}M^{(i)} = \mathbf{1}_{256}^{\mathrm{T}} > 0.$$

根据推论 10.2.1 可知, 该输出跟踪问题可解.

再利用命题 10.3.4 可知, 这里有 $2^{203} \times 3^{59}$ 个可行的反馈增益矩阵, 它们可依如下规定选择:

$$\mathrm{Col}_i(K) \in \begin{cases} \delta_4\{1,2,3,4\},\ i = 8, 16, 23, 24, 29-32, 40, 48, 55,\\ \qquad 56, 61-64, 72, 80, 87, 88, 93-96,\\ \qquad 99, 104, 112, 119, 120, 125-128,\\ \delta_4\{1,2,4\},\ i = 33-38, 41-44, 49-54,\\ \qquad 57-60, 97, 98, 100-102, 105-108,\\ \qquad 113-118, 121-124, 161-166,\\ \qquad 169-172, 177-182, 185-188,\\ \delta_4\{3,4\},\ i = 7, 13-15, 71, 77-79, 135,\\ \qquad 136, 141-144, 151, 152, 157-160, 199,\\ \qquad 200, 205-208, 215, 216, 221-224, 231,\\ \qquad 232, 240, 247, 248, 253-256,\\ \delta_4\{3\},\ i = 39, 45-47, 103, 109-111, 167,\\ \qquad 168, 173-176, 183, 184, 189-192, 237-239,\\ \delta_4\{4\},\ i = 229, 230, 245, 246,\\ \delta_4\{1,2\},\ \text{其他}. \end{cases}$$

例如选 $(x(0),\hat{z}(0))=\delta_{256}^1$. 图 10.3.2 刻画了跟踪过程, 这里, 当 $(x(0),\hat{z}(0))=\delta_{256}^1$ 时, 只要 K 按上述原则选取, $y_1(t)=\hat{y}(t)=\delta_2^1$, $t\geqslant 2$, 所有的 k 会导致如图 10.3.2 所示轨线.

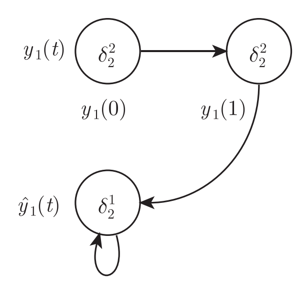


图 10.3.2　输出跟踪

第 11 章　概率布尔网络

概率布尔网络及概率布尔控制网络是布尔网络理论中较艰深而又十分重要的一个部分, 除了逻辑动态系统的基本知识外, 它还与随机过程, 特别是马尔可夫链 (简称马氏链) 密切相关. 基于矩阵半张量积的代数状态空间方法也被广泛应用于概率布尔网络及其控制的研究中, 目前研究的热点主要集中于能控能观性[48,86,140,141] 及镇定控制[57,90,152,153] 等.

11.1　马尔可夫链简介

11.1.1　概率转移矩阵

因为随机布尔网络的研究依赖于马尔可夫链, 本节对此作一简单回顾. 更多细节参见相关参考文献, 例如 [10,16].

设 $X(t)$, $t=0,1,2,\cdots$ 为一个随机过程, $X(t)\in\{1,2,\cdots,k\}$ (通常 k 亦可为 ∞), 这里我们只关心有限个值的情况. 如果 $X(t)$ 在 $t+1$ 时刻的值只依赖于它在 t 时刻的值, 即

$$
\begin{aligned}
&\mathbb{P}(X(t+1)=j_{t+1}\mid X(t)=j_t,X(t-1)=j_{t-1},\cdots,X(1)=j_1,X(0)=j_0)\\
=\;&\mathbb{P}(X(t+1)=j_{t+1}\mid X(t)=j_t),
\end{aligned}
$$

则称 $X(t)$, $t=0,1,2,\cdots$ 为一个马尔可夫链.

一般假定马尔可夫链是齐次的, 即

$$
\mathbb{P}(X(t+1)=i\mid X(t)=j)=p_{ij},\quad i,j=1,2,\cdots,k, \tag{11.1.1}
$$

这里, p_{ij} 与 t 无关.

在 k 值逻辑系统里, 一个逻辑变量 X 可取 k 个值, 我们将其记作 $X\in\Delta_k$. 在一个可取 k 个值的马尔可夫链里, 设 $X(t)$ 取 k 个值的概率分别为 r_i, $i=1,2,\cdots,k$, 则 $r_i\geqslant 0$, $\sum_{i=1}^k r_i=1$. 记

$$
\Upsilon_k:=\left\{r\in\mathbb{R}^k\;\middle|\;r_i\geqslant 0,\ \sum_{i=1}^k r_i=1\right\}.
$$

则得 $x(t) \in \Upsilon_k$. 类似于 k 值逻辑变量的向量表示, 我们称 $x(t)$ 为 $X(t)$ 的向量表示, 记作 $x(t) = \vec{X}(t)$. 实际上

$$x(t) = \begin{bmatrix} r_1(t) \\ r_2(t) \\ \vdots \\ r_k(t) \end{bmatrix} = \sum_{i=1}^{k} r_i(t)\delta_k^i. \tag{11.1.2}$$

因此, 这种表达式可视作 $X(t)$ 的期望值.

将转移概率放到一起, 则得到概率转移矩阵:

$$P := \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1k} \\ p_{21} & p_{22} & \cdots & p_{2k} \\ \vdots & \vdots & & \vdots \\ p_{k1} & p_{k2} & \cdots & p_{kk} \end{bmatrix}.$$

这里, $\mathrm{Col}(P) \subset \Upsilon_k$.

那么, 显然

$$x(t) = P^t x(0). \tag{11.1.3}$$

P^t 称为 t 步概率转移矩阵.

11.1.2　状态分类

考虑一个马氏链 $X(t)$, $t = 0, 1, 2, \cdots$. 设它的取值范围是 $N = \{1, 2, \cdots, k\}$.

(i) 如果 $x(t) \in \Delta_k$, 则称为它的一个纯状态.

(ii) 如果 $x(t) \in \Upsilon_k$, 则称为它的一个混合状态.

以下的讨论只针对纯状态.

定义 11.1.1　设 $i, j \in N$.

(i) 称状态 i 可达 (communicate with) 状态 j, 记作 $i \to j$, 如果存在 $t \geqslant 1$, 使得 $p_{i,j}^{(t)} > 0$.

(ii) 称状态 i 与状态 j 相通 (intercommunicate with) , 记作 $i \leftrightarrow j$, 如果 $i \to j$ 且 $j \to i$.

(iii) N 称为全连通的, 如果 $i \leftrightarrow j$, $\forall i, j \in N$.

命题 11.1.1　设 $i \to j$, $j \to k$, 则 $i \to k$.

证明　由定义可知, 存在 $\alpha, \beta \geqslant 1$ 使得 $p_{i,j}^{(\alpha)} > 0$, $p_{j,k}^{(\beta)} > 0$. 由马尔可夫性可知

$$p_{i,k}^{\alpha+\beta} = \sum_{s=1}^{n} p_{i,s}^{\alpha} p_{s,k}^{\beta} \geqslant p_{i,j}^{\alpha} p_{j,k}^{\beta} > 0.$$

于是有 $i \to k$. □

定义 11.1.2 设 $i,j \in N$.

(i) 记 $r_{i,j}$ 为从 i 到 j 的首次到达时间 (首达时间), 即

$$r_{i,j}=\begin{cases}\min\{k \mid x(0)=i, x(k)=j\}, & \text{其他}\\ \infty, & \{k \mid x(0)=i, x(k)=j\}=\varnothing.\end{cases} \tag{11.1.4}$$

(ii) 记首达时间等于 s 的条件概率为

$$f_{i,j}^{(s)} := \mathbb{P}\left(r_{i,j}=s \mid x(0)=i\right). \tag{11.1.5}$$

(iii) 系统从 i 出发, 经有穷步到达 j 的条件概率为

$$f_{i,j} := \sum_{s=1}^{\infty} f_{i,j}^{(s)} = \mathbb{P}(r_{i,j}<\infty). \tag{11.1.6}$$

下面的命题指出了首达概率与转移概率的关系.

命题 11.1.2 设 $i,j \in N$, 则

$$p_{i,j}^{(s)} = \sum_{\ell=1}^{s} f_{i,j}^{(\ell)} p_{j,j}^{(k-\ell)}. \tag{11.1.7}$$

下面的命题指出了有穷步到达与可达的关系.

命题 11.1.3 设 $i,j \in N$, 则 $i \to j$ 当且仅当 $f_{i,j}>0$.

下面的命题是状态分类的基础.

命题 11.1.4 令 $i \in N$. 如果 $f_{i,i}=1$, 则系统从 i 出发, 依概率 1 无数次返回 i; 如果 $f_{i,i}<1$, 则系统从 i 出发, 依概率 1 只有有限次返回 i.

定义 11.1.3 令 $i \in N$. 如果 $f_{i,i}=1$, 则称状态 i 是常返的 (recurrent); 如果 $f_{i,i}<1$, 则称状态 i 是过渡的 (transient).

常返也可由转移概率来判定.

命题 11.1.5 令 $i \in N$. 状态 i 常返的充要条件是

$$\sum_{s=1}^{\infty} p_{i,i}^{(s)} = \infty.$$

设 i 是常返的, 那么 i 的平均返回时间 (期望值) 为

$$\mu_i = \sum_{k=1}^{\infty} k \cdot f_{i,i}^{(k)}. \tag{11.1.8}$$

定义 11.1.4 (马氏链状态的周期性、遍历性) (i) 设 $i \in N$ 是常返的. 如果 $\mu_i < \infty$, 则称 i 为正常返的 (positive recurrent); 如果 $\mu_i = \infty$, 则称 i 为零常返的 (null recurrent).

(ii) 令 $i \in N$. 设 $\{k \mid p_{i,i}^{(k)} > 0\}$ 的最大公约数为 s. 如果 $s \geqslant 2$, 则称 i 为周期的或 s 周期的 (periodic); 如果 $s = 1$, 则称 i 为非周期的 (non-periodic).

(iii) 令 $i \in N$. 如果 i 是正常返、非周期的, 则称 i 是遍历的 (ergodic).

以下命题揭示了转移概率与状态性质的关系.

命题 11.1.6 (i) 设 $i \in N$ 是常返的, 则 i 为零常返的充要条件是

$$\lim_{n\to\infty} p_{i,i}^{(n)} = 0. \tag{11.1.9}$$

(ii) 设 $i \in N$ 是正常返的, 则

$$\limsup_{n\to\infty} p_{i,i}^{(n)} > 0. \tag{11.1.10}$$

特别是:

如果 i 是遍历的, 则

$$\lim_{n\to\infty} p_{i,i}^{(n)} = \frac{1}{\mu_i}. \tag{11.1.11}$$

如果 i 是 ℓ 周期的, 则

$$\lim_{n\to\infty} p_{i,i}^{(\ell n)} = \frac{\ell}{\mu_i}. \tag{11.1.12}$$

以下命题说明, 如果两状态相通, 则它们属于同种类型.

命题 11.1.7 设 $i \leftrightarrow j$, 则

(i) 它们同为非常返或同为常返;

(ii) 如果它们同为常返, 则它们同为零常返或同为正常返;

(iii) 它们同为非周期的或同为周期的;

(iv) 如果它们同为周期的, 则它们的周期也相同.

例 11.1.1 一个马氏链, $N = \{1, 2, 3\}$. 其状态转移矩阵为

$$P = \begin{bmatrix} \frac{1}{2} & 0 & 1 \\ \frac{1}{4} & 0 & 0 \\ \frac{1}{4} & 1 & 0 \end{bmatrix}.$$

(1)　状态 1 是否常返? 正常返? 周期或遍历?

(2)　状态 2 及状态 3 是否常返? 正常返? 周期或遍历?

我们计算如下:

$$
\begin{aligned}
f_{1,1}^{(1)} &= \frac{1}{2}. \\
f_{1,1}^{(2)} &= \mathbb{P}\left(x(2)=1, x(1) \neq 1 | x(0)=1\right) \\
&= \mathbb{P}\left(x(2)=1, x(1)=2 | x(0)=1\right) \\
&\quad + \mathbb{P}\left(x(2)=1, x(1)=3 | x(0)=1\right) \\
&= p_{1,2} \cdot p_{2,1} + p_{1,3} \cdot p_{3,1} \\
&= \frac{1}{4}. \\
f_{1,1}^{(3)} &= \mathbb{P}\left(x(3)=1, x(1) \neq 1, x(2) \neq 1 | x(0)=1\right) \\
&= \mathbb{P}\left(x(3)=1, x(1)=2, x(2)=2 | x(0)=1\right) \\
&\quad + \mathbb{P}\left(x(3)=1, x(1)=2, x(2)=3 | x(0)=1\right) \\
&\quad + \mathbb{P}\left(x(3)=1, x(1)=3, x(2)=2 | x(0)=1\right) \\
&\quad + \mathbb{P}\left(x(3)=1, x(1)=3, x(2)=3 | x(0)=1\right) \\
&= 0 + \frac{1}{4} + 0 + 0 = \frac{1}{4}.
\end{aligned}
$$

于是有

$$
f_{1,1} = \frac{1}{2} + \frac{1}{4} + \frac{1}{4} = 1.
$$

因此 1 是常返的.

$$
\mu_1 = 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{4} + 3 \cdot \frac{1}{4} = \frac{7}{4} < \infty.
$$

由命题 11.1.6 可知, 1 是正常返的. 因 $f_{1,1}^{(1)} = \dfrac{1}{2} > 0$, 可知, 1 是非周期的, 因此是遍历的.

进而, 由概率转移矩阵可知, $1 \to 2 \to 3 \to 1$. 由命题 11.1.1 可知, $1 \leftrightarrow 2 \leftrightarrow 3$. 由命题 11.1.7 可知, 2 和 3 也是遍历的.

11.1.3　随机矩阵的收敛性

定义 11.1.5 (马氏链的可约性)　(i)　设 $X(t)$ 为 N 上的一个马氏链, $C \subset N$. C 称为一个连通集, 如果对任意 $i, j \in C$, 均有 $i \leftrightarrow j$.

C 称为一个闭集, 如果对任意 $i \in C$, $\ell \in C^c$, 均有 $p_{i,\ell} = 0$. 这里 C^c 代表 C 的余集, 即 $C^c = \{x \in N \mid x \notin C\}$.

(ii) 一个马氏链, 如果除空集 ($\varnothing$) 与全空间 (N) 外没有其他闭集, 则称其为不可约的.

(iii) 一个闭集称为不可约集, 如果它不包括一个非空真闭子集.

(iv) 一个随机矩阵称为不可约的, 如果它所定义的马氏链不可约.

注意, 空集可以看作闭集, 全连通的全空间也是闭集. 其他闭集 C ($C \neq \varnothing$, $C \neq N$) 称为非平凡闭集.

经过状态重新排序, 可设 $C = \{1, 2, \cdots, s\}$ 及 $C^c = \{s+1, s+2, \cdots, k\}$. 那么, C 是闭集表示状态的概率转移矩阵可表示为下三角形式:

$$P = \begin{bmatrix} P_C & 0 \\ * & * \end{bmatrix}, \tag{11.1.13}$$

这里 $P_C \in \mathcal{M}_{s\times s}$; 并且, 存在 $t > 0$, 使得

$$P_C^t > \mathbf{1}_k.$$

设一个马氏链的状态空间为 $N = \{1, 2, \cdots, k\}$, 则它有一个剖分

$$N = C^c \cup C_1 \cup C_2 \cup \cdots \cup C_s.$$

这里 C^c 是过渡状态. $C = C_1 \cup C_2 \cup \cdots \cup C_s$ 为 (正) 常返状态, 其中 C_i, $i = 1, 2, \cdots, s$ 为不可约闭集.

定义 11.1.6　一个矩阵 P 称为本原的, 如果它是非负的而且存在一个整数 $s > 0$ 使 P^s 所有元素为正的.

本原矩阵也称为素矩阵. 可以证明如下定理.

定理 11.1.1　本原矩阵等价于不可约非周期非负矩阵.

随机矩阵是特殊的非负矩阵, 其本原性有如下的等价定义.

定义 11.1.7　一个随机矩阵 P 称为本原的, 如果

(i) 它是不可约的, 并且,

(ii) 仅有一个特征值 $\lambda \in \sigma(P)$ 满足 $|\lambda| = 1$.

下面的命题用于检验随机矩阵的本原性是很方便的.

命题 11.1.8　随机矩阵 P 是本原的, 当且仅当存在 $k > 0$, 使 P^k 所有元素为正.

随机矩阵的收敛性是很重要的. 考察式 (11.1.3). 如果

$$\lim_{k\to\infty} P^k := P_\infty \tag{11.1.14}$$

存在, 则对每个 x_0 可知它相应的稳态分布

$$x(\infty) = P_\infty x_0, \tag{11.1.15}$$

其中 $x_0 = x(0)$.

下面的定理十分有用.

定理 11.1.2 [62]　设一个随机矩阵 P 为本原的, 那么

$$\lim_{k\to\infty} P^k := L > 0, \tag{11.1.16}$$

这里 $L = yx^{\mathrm{T}}$, x, y 满足: $x^{\mathrm{T}}P = x^{\mathrm{T}}$, $x > 0$, $Py = y$, $y > 0$, 且 $x^{\mathrm{T}}y = 1$.

例 11.1.2　一个随机矩阵 P 为

$$P = \begin{bmatrix} 0.2 & 0.5 & 0 \\ 0.4 & 0 & 0.6 \\ 0.4 & 0.5 & 0.4 \end{bmatrix}.$$

P 不可约, 因为 $1 \leftrightarrow 2$ 且 $2 \leftrightarrow 3$. 它没有非平凡闭集. 计算可知, 其特征值为 $\sigma(P) = \{1, 0.4449, 0.0449\}$, 故模为 1 的特征值唯一. 根据定义, 它是本原的. 根据定理 11.1.2, P^k $(k \to \infty)$ 收敛. 可算出 $n = 44$ 时它收敛, 于是近似有

$$P^{44} \approx P^{45} = \begin{bmatrix} 0.2174 & 0.2174 & 0.2174 \\ 0.3478 & 0.3478 & 0.3478 \\ 0.4348 & 0.4348 & 0.4348 \end{bmatrix}. \tag{11.1.17}$$

实际上, 满足 $Py = y$, $y > 0$ 的解可取 $y = \mathbf{1}_3$. 即 $x = (1,1,1)^{\mathrm{T}}$. 那么, 容易得到, 满足 $Py = y$, $y > 0$, 且 $x^{\mathrm{T}}y = 1$ 的 y 为

$$y = [0.2174, 0.3478, 0.4348]^{\mathrm{T}}.$$

于是可知, P^t $(t \to \infty)$ 收敛于 $L = yx^{\mathrm{T}}$, 它就是式 (11.1.17) 中的那个矩阵.

注意, 在上例中, 极限矩阵 L 各列都一样. 上述例子中最后的分析告诉我们, 如果 P 是本原的, 它的极限矩阵一定有这个性质. 同时, 这个性质非常重要. 设 P 是本原的, 由式 (11.1.15) 有

$$x(\infty) = P_{\infty}x_0 = Lx_0.$$

由于 L 每一列都一样, 这表明, 无论初始状态 x_0 是怎样的, 马氏链最后都会趋于一个相同的稳态分布.

11.2　概率布尔网络的矩阵表示

考察一个布尔网络

$$\begin{cases} X_1(t+1) = F^1(X_1(t), X_2(t), \cdots, X_n(t)), \\ X_2(t+1) = F^2(X_1(t), X_2(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F^n(X_1(t), X_2(t), \cdots, X_n(t)). \end{cases} \tag{11.2.1}$$

定义 11.2.1 [112] (11.2.1) 称为一个概率布尔网络, 如果其中的逻辑函数 F^i 可能有多个模型. 严格地说,

$$F^i \in \left\{F_1^i, F_2^i, \cdots, F_{\ell_i}^i\right\}, \quad i = 1, 2, \cdots, n;$$

并且,

$$\mathbb{P}(F^i = F_j^i) = p_{ij}, \quad j = 1, 2, \cdots, n, \quad i = 1, 2, \cdots, \ell_i, \tag{11.2.2}$$

这里,

$$\sum_{j=1}^{\ell_i} p_{ij} = 1, \quad i = 1, 2, \cdots, n.$$

注意到一个概率布尔网络可有 $\ell = \prod_{i=1}^n \ell_i$ 个不同的模型, 我们用一个指标矩阵 K 来刻画它.

$$K = \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & 1 & \cdots & 1 & 2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & \ell_n \\ 1 & 1 & \cdots & 2 & 1 \\ 1 & 1 & \cdots & 2 & 2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \cdots & 2 & \ell_n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \ell_1 & \ell_2 & \cdots & \ell_{n-1} & \ell_n \end{bmatrix}. \tag{11.2.3}$$

这里, $K \in \mathcal{M}_{N\times n}$, 其中, $N = \prod_{j=1}^n \ell_j$.

于是, K 的每一行代表一个模型, 而概率布尔网络实现这个模型的概率为

$$p_i = \mathbb{P}(\text{模型 } i \text{ 实现}) = \prod_{j=1}^n p_{i,K_{ij}}. \tag{11.2.4}$$

记 $x_i = \vec{X}_i$, $i = 1, 2, \cdots, n$, $x(t) := \ltimes_{i=1}^n x_i(t)$. 那么对每一个模型可构造其代数状态空间表达式

$$x(t+1) = M_i x(t), \quad i = 1, 2, \cdots, N. \tag{11.2.5}$$

因此可得概率布尔网络的代数状态空间表达式

$$x(t+1) = \sum_{i=1}^N p_i M_i x(t) := M x(t), \tag{11.2.6}$$

这里,

$$M := \sum_{i=1}^N p_i M_i \in \Upsilon_{2^n \times 2^n}$$

称为概率布尔网络的结构矩阵. 因为 M 是个概率转移矩阵, 概率布尔网络的的状态轨线就是一个马尔可夫链. 因此, 关于马尔可夫链的性质对它都成立, 例如, 下面这个结论是重要的.

命题 11.2.1 设概率布尔网络 (11.2.1) 的结构矩阵 M 为本原矩阵, 则对每一个变量 X_i, 存在一个稳态分布, 即, 存在 (p_1^i, p_0^i), $p_1^i \geqslant 0$, $p_0^i \geqslant 0$, 且 $p_1^i + p_0^i = 1$, 使得

$$\begin{cases} \mathbb{P}(X_i(\infty) = 1) = p_1^i, \\ \mathbb{P}(X_i(\infty) = 0) = p_0^i, \quad i = 1, 2, \cdots, n. \end{cases} \tag{11.2.7}$$

证明 因为 M 为本原矩阵, 根据定理 11.1.2, 则存在 $\xi \in \Upsilon_{2^n}$, 使得

$$x(\infty) = \xi.$$

于是

$$x_i(\infty) = \left(\mathbf{1}_{2^{i-1}}^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_{2^{n-i}}^{\mathrm{T}} \xi\right) := \begin{bmatrix} p_1^i \\ p_0^i \end{bmatrix}, \quad i = 1, 2, \cdots, n. \tag{11.2.8}$$

□

下面给出一个例子.

例 11.2.1 考察如下概率布尔网络:

$$\begin{cases} X_1(t+1) = F^1(X_1(t), X_2(t), X_3(t)), \\ X_2(t+1) = F^2(X_1(t), X_2(t), X_3(t)), \\ X_3(t+1) = F^3(X_1(t), X_2(t), X_3(t)), \end{cases} \tag{11.2.9}$$

这里,

$$\begin{cases} F_1^1(X(t)) = X_1(t) \wedge X_2(t), \\ F_2^1(X(t)) = X_2(t) \bar{\vee} X_3(t), \\ F_3^1(X(t)) = X_1(t) \vee (X_2(t) \leftrightarrow X_3(t)) \end{cases}$$

且

$$\mathbb{P}(F^1 = F_1^1) = 0.4, \quad \mathbb{P}(F^1 = F_2^1) = \mathbb{P}(F^1 = F_3^1) = 0.3.$$

$$\begin{cases} F_1^2(X(t)) = X_1(t) \bar{\vee} X_2(t), \\ F_2^2(X(t)) = X_2(t) \to (\neg X_3(t)) \end{cases}$$

且

$$\mathbb{P}(F^2 = F_1^2) = 0.3, \quad \mathbb{P}(F^2 = F_2^2) = 0.7.$$

$$\begin{cases} F_1^3(X(t)) = X_1(t) \wedge X_3(t), \\ F_2^3(X(t)) = X_1(t) \leftrightarrow (X_2(t) \wedge X_3(t)) \end{cases}$$

且

$$\mathbb{P}(F^3 = F_1^3) = \mathbb{P}(F^2 = F_2^3) = 0.5.$$

于是, 指标矩阵 K 如下:

$$K = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 2 \\ 1 & 2 & 1 \\ 1 & 2 & 2 \\ 2 & 1 & 1 \\ 2 & 1 & 2 \\ 2 & 2 & 1 \\ 2 & 2 & 2 \\ 3 & 1 & 1 \\ 3 & 1 & 2 \\ 3 & 2 & 1 \\ 3 & 2 & 2 \end{bmatrix}.$$

其相应的概率为

$$p_1 = 0.4 \times 0.3 \times 0.5 = 0.06,$$
$$p_2 = 0.4 \times 0.3 \times 0.5 = 0.06,$$
$$p_3 = 0.4 \times 0.7 \times 0.5 = 0.14,$$

$$
\begin{aligned}
p_4 &= 0.4 \times 0.7 \times 0.5 = 0.14,\\
p_5 &= 0.3 \times 0.3 \times 0.5 = 0.045,\\
p_6 &= 0.3 \times 0.3 \times 0.5 = 0.045,\\
p_7 &= 0.3 \times 0.7 \times 0.5 = 0.105,\\
p_8 &= 0.3 \times 0.7 \times 0.5 = 0.105,\\
p_9 &= 0.3 \times 0.3 \times 0.5 = 0.045,\\
p_{10} &= 0.3 \times 0.3 \times 0.5 = 0.045,\\
p_{11} &= 0.3 \times 0.7 \times 0.5 = 0.105,\\
p_{12} &= 0.3 \times 0.7 \times 0.5 = 0.105.
\end{aligned}
$$

记第 i 个模态的结构矩阵为 M_i. 不难算得

$$
\begin{aligned}
M_1 &= \delta_8[3,4,5,6,6,6,8,8],\\
M_2 &= \delta_8[3,4,6,6,6,5,7,7],\\
M_3 &= \delta_8[3,2,5,6,8,6,6,6],\\
M_4 &= \delta_8[3,2,6,6,8,5,5,5],\\
M_5 &= \delta_8[3,4,1,6,2,2,4,8],\\
M_6 &= \delta_8[3,4,5,6,6,6,8,8],\\
M_7 &= \delta_8[3,2,1,6,4,2,2,6],\\
M_8 &= \delta_8[3,2,2,6,4,1,1,5],\\
M_9 &= \delta_8[3,4,1,2,2,6,8,4],\\
M_{10} &= \delta_8[3,4,2,2,2,5,7,3],\\
M_{11} &= \delta_8[3,2,1,2,4,6,6,2],\\
M_{12} &= \delta_8[3,2,2,2,4,5,5,1].
\end{aligned}
$$

于是, 概率布尔网络 (11.2.9) 的结构矩阵为

$$
M = \sum_{i=1}^{12} p_i M_i = \begin{bmatrix}
0 & 0 & 0.3 & 0 & 0 & 0.15 & 0.105 & 0.105\\
0 & 0.7 & 0.3 & 0.3 & 0.18 & 0.15 & 0.105 & 0.105\\
1 & 0 & 0 & 0 & 0 & 0 & 0.045 & 0.045\\
0 & 0.3 & 0 & 0 & 0.42 & 0 & 0.045 & 0.045\\
0 & 0 & 0.2 & 0 & 0 & 0.35 & 0.245 & 0.245\\
0 & 0 & 0.2 & 0.7 & 0.12 & 0.35 & 0.245 & 0.245\\
0 & 0 & 0 & 0 & 0 & 0 & 0.105 & 0.105\\
0 & 0 & 0 & 0 & 0.28 & 0 & 0.105 & 0.105
\end{bmatrix}. \tag{11.2.10}
$$

容易检验 $M^4 > 0$, 因此这个结构矩阵是本原的. 计算可知

$$M^t \to M_\infty, \quad t \to \infty,$$

这里

$$M_\infty = \begin{bmatrix} 0.0523 & 0.0523 & \cdots & 0.0523 \\ 0.3896 & 0.3896 & \cdots & 0.3896 \\ 0.0538 & 0.0538 & \cdots & 0.0538 \\ 0.1583 & 0.1583 & \cdots & 0.1583 \\ 0.0951 & 0.0951 & \cdots & 0.0951 \\ 0.2173 & 0.2173 & \cdots & 0.2173 \\ 0.0035 & 0.0035 & \cdots & 0.0035 \\ 0.0302 & 0.0302 & \cdots & 0.0302 \end{bmatrix}.$$

因此,

$$X(t) \to \xi = [0.0523, 0.3896, 0.0538, 0.1583, 0.0951, 0.2173, 0.0035, 0.0302]^{\mathrm{T}}.$$

于是

$$x_1(\infty) = (I_2 \otimes \mathbf{1}_4^{\mathrm{T}})\xi = \begin{bmatrix} 0.6539 \\ 0.3461 \end{bmatrix},$$

$$x_2(\infty) = (\mathbf{1}_2^{\mathrm{T}} \otimes I_2 \otimes \mathbf{1}_4^{\mathrm{T}})\xi = \begin{bmatrix} 0.7542 \\ 0.2458 \end{bmatrix},$$

$$x_3(\infty) = (\mathbf{1}_4^{\mathrm{T}} \otimes I_2)\xi = \begin{bmatrix} 0.2047 \\ 0.7953 \end{bmatrix}.$$

也就是说

$$\mathbb{P}(X_1(\infty) = 1) = 0.6539, \quad \mathbb{P}(X_1(\infty) = 0) = 0.3461.$$
$$\mathbb{P}(X_2(\infty) = 1) = 0.7542, \quad \mathbb{P}(X_2(\infty) = 0) = 0.2458.$$
$$\mathbb{P}(X_3(\infty) = 1) = 0.2047, \quad \mathbb{P}(X_3(\infty) = 0) = 0.7953.$$

考察一个布尔控制网络

$$\begin{cases} X_1(t+1) = F^1(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), U_2(t), \cdots, U_m(t)), \\ X_2(t+1) = F^2(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), U_2(t), \cdots, U_m(t)), \\ \qquad \cdots\cdots \\ X_n(t+1) = F^n(X_1(t), X_2(t), \cdots, X_n(t), U_1(t), U_2(t), \cdots, U_m(t)). \end{cases} \tag{11.2.11}$$

如果其中的逻辑函数 F^i 可能有多个模型, 即

$$F^i \in \left\{F_1^i, F_2^i, \cdots, F_{\ell_i}^i\right\}, \quad i=1,2,\cdots,n,$$

并且,

$$\mathbb{P}(F^i = F_j^i) = p_{ij}, \quad j=1,2,\cdots,n, \quad i=1,2,\cdots,\ell_i, \tag{11.2.12}$$

这里,

$$\sum_{j=1}^{\ell_i} p_{ij} = 1, \quad i=1,2,\cdots,n,$$

则 (11.2.11) 称为一个概率布尔控制网络.

因此, 一个概率布尔控制网络有 $N = \prod_{i=1}^n \ell_i$ 个模态, 记作 Σ_λ, $\lambda \in \Lambda = \{1,2,\cdots,N\}$. 每一个 λ 对应一组 $\left(j_1^\lambda, j_2^\lambda, \cdots, j_n^\lambda\right)$, 这里 $1 \leqslant j_i^\lambda \leqslant \ell_i$, 表明

$$\Sigma_\lambda = \left(F_{j_1^\lambda}^1, F_{j_2^\lambda}^2, \cdots F_{j_n^\lambda}^n\right).$$

于是, 布尔控制网络 (11.2.11) 取模态 Σ_λ 的概率为

$$p_\lambda = \prod_{i=1}^n p_{\lambda, j_i^\lambda}, \quad \lambda \in \Lambda.$$

定义 11.2.2 一个矩阵 $L \in \mathcal{M}_{p\times q}$ 称为一个随机矩阵, 如果存在一族矩阵 $\{L_1, L_2, \cdots, L_N\} \subset \mathcal{M}_{p\times q}$ 使得

$$\mathbb{P}(L = L_i) = p_i, \quad i=1,2,\cdots,N, \tag{11.2.13}$$

这里, $\sum_{i=1}^n p_i = 1$.

布尔控制网络 (11.2.11) 的代数状态空间表示为

$$x(t+1) = L(t)u(t)x(t), \tag{11.2.14}$$

这里, $L(1)$, $L(2)$, $\cdots$ 是一列独立同分布随机逻辑矩阵.

它的另一种表达是

$$x(t+1) = Lu(t)x(t), \tag{11.2.15}$$

这里

$$L = \sum_{i=1}^N p_i L_i.$$

(11.2.15) 实际上表示的是期望值的演化. 为了区别, 称 (11.2.14) 为布尔控制网络的随机代数状态空间表示 , 称 (11.2.15) 为布尔控制网络的期望值代数状态空间表示.

例 11.2.2　考察布尔控制网络

$$\begin{cases} X_1(t+1) = F^1(X_2(t), U(t)), \\ X_2(t+1) = F^2(X_1(t), X_2(t)), \end{cases} \tag{11.2.16}$$

这里,

$$\begin{aligned} & F^1 \in \{F_1^1, F_2^1, F_3^1\}, \quad F^2 \in \{F_1^2, F_2^2\}, \\ & F_1^1(X_2, U) = X_2 \bar{\vee} U, \\ & F_2^1(X_2, U) = X_2 \vee U, \\ & F_3^1(X_2, U) = X_2 \wedge U, \\ & F_1^2(X_1, X_2) = X_1 \wedge X_2, \\ & F_2^2(X_1, X_2) = X_1 \to X_2. \end{aligned}$$

并且 $\mathbb{P}(F_1 = F_i^1) = \dfrac{1}{3}$, $i = 1, 2, 3$, $\mathbb{P}(F^2 = F_i^2) = \dfrac{1}{2}$, $i = 1, 2$.

于是有

$$\begin{aligned} & \Sigma_1 = (F_1^1, F_1^2), \quad \Sigma_2 = (F_1^1, F_2^2), \quad \Sigma_3 = (F_1^1, F_3^2), \\ & \Sigma_4 = (F_2^1, F_1^2), \quad \Sigma_5 = (F_2^1, F_2^2), \quad \Sigma_6 = (F_2^1, F_3^2). \end{aligned}$$

相应的结构矩阵为

$$\begin{aligned} & L_1 = \delta_4[3, 2, 4, 2, 1, 4, 2, 4], \quad L_2 = \delta_4[3, 2, 3, 1, 1, 4, 1, 3], \\ & L_3 = \delta_4[1, 2, 2, 2, 1, 4, 2, 4], \quad L_4 = \delta_4[1, 2, 1, 1, 1, 4, 1, 3], \\ & L_5 = \delta_4[1, 4, 2, 4, 3, 4, 4, 4], \quad L_6 = \delta_4[1, 4, 1, 3, 3, 4, 3, 3]. \end{aligned} \tag{11.2.17}$$

在随机代数状态空间表示中

$$\mathbb{P}(L(t) = L_i) = \frac{1}{6}, \quad i = 1, 2, 3, 4, 5, 6.$$

在期望代数状态空间表示中

$$L = \frac{1}{6} \begin{bmatrix} 4 & 0 & 2 & 2 & 4 & 0 & 2 & 0 \\ 0 & 4 & 2 & 2 & 0 & 0 & 2 & 0 \\ 2 & 0 & 1 & 1 & 2 & 0 & 1 & 3 \\ 0 & 2 & 1 & 1 & 0 & 6 & 1 & 3 \end{bmatrix}.$$

11.3 概率布尔网络的能控性

定义 11.3.1 考察概率布尔控制网络 (11.2.11).

(i) 该网络可由 X_i (依概率 1) 控制到 X_j, 如果存在一个控制序列, 使得

$$\mathbb{P}\left(X(t)=X_j \text{对某个 } t \geqslant 1 \mid X(0)=X_i\right)=1. \tag{11.3.1}$$

(ii) 该网络在 X_i 能控, 如果对任何 $X_j \in \mathcal{D}^n$, 由 X_i 到 X_j 能控,

(iii) 该网络能控, 如果对任何 $X_i, X_j \in \mathcal{D}^n$, 由 X_i 到 X_j 均能控.

为解决能控性问题, 我们引入联合可达的概念.

定义 11.3.2 考察概率布尔控制网络 (11.2.11).

(i) X_j 称为由 X_i 联合可达的, 如果存在一个模型–控制对序列 $\{(\Sigma(t), u(t)), t=0,1,2,\cdots,s-1\}$, $0<s<\infty$, 使得受控的网络轨线在某个时刻 s 到达 x_d, 这里在每个时刻 t 被选的模型为 $\Sigma(t)$, 相应的控制为 $u(t)$.

(ii) X_j 称为联合可达的, 如果对任何 $X_i \in \mathcal{D}^n$, 它都是由 X_i 联合可达的.

(iii) 网络 (11.2.1) 称为联合可达的, 如果对任何 $X_i, X_j \in \mathcal{D}^n$, X_j 都是由 X_i 联合可达的.

不难看出, 如果网络是能控的, 它也一定是联合可达的, 下面我们证明: 能控性与联合可达等价.

我们先寻找可达的一个等价的可计算条件, 再证明能控性与联合可达的等价性, 将这两部分合到一起, 就可以解决能控性问题.

记 $\Lambda=\{1,2,\cdots,N\}$ 为所有模型的指标集, 这里 $N=\prod_{j=1}^{n}\ell_j$. 令 $\lambda \in \Lambda$, 则 Σ_λ 为一确定性模型, L_λ 为其结构矩阵, 定义

$$M_\lambda=\sum_{j=1}^{2^m}{}_{\mathcal{B}}\operatorname{Blk}_j(L_\lambda), \tag{11.3.2}$$

$$M_P=\sum_{\lambda=1}^{N}{}_{\mathcal{B}}M_\lambda:=[m_{i,j}]. \tag{11.3.3}$$

那么, M_P 的 i,j 位元素 $m_{i,j}=1$ 意味着至少存在一个模态 Σ 和一个控制 u 使得 $\delta_{2^n}^i$ 可以从 $\delta_{2^n}^j$ 出发一步到达. 于是我们可以定义一个 $2^n \times 2^n$ 布尔矩阵, 称为能达矩阵, 如下:

$$\mathcal{R}_P=\sum_{s=1}^{2^n}{}_{\mathcal{B}}M_P^{(s)} \in \mathcal{B}_{2^n\times 2^n}. \tag{11.3.4}$$

下面的定理表明, $\mathcal{R}_P$ 可以完全刻画联合可达性.

定理 11.3.1 考察概率布尔控制网络 (11.2.1). 它有 N 个模态, 记作 $\Sigma_\lambda, \lambda \in \Lambda = \{1, 2, \cdots, N\}$. 该网络联合可达, 当且仅当,

$$\mathcal{R}_P > 0.$$

证明 (充分性) 首先证明: 如果 $(M_P^{(s)})_{j,i} > 0$, 则从 $\delta_{2^n}^i$ 出发可在 s 时刻到达 $\delta_{2^n}^j$. 我们用数学归纳法证明. 当 $s = 1$ 时, 结论显见. 设 $s = k$ 时结论成立, 那么, 当 $s = k + 1$ 时, 从 $\delta_{2^n}^i$ 在时间 $k + 1$ 时到达 $\delta_{2^n}^j$, 当且仅当, 存在 $\delta_{2^n}^\alpha$ 使得 $\delta_{2^n}^j$ 是从 $\delta_{2^n}^\alpha$ 在 k 时间出发, 而在 $k + 1$ 时间到达的. 因此, $(M_P)_{j,\alpha} > 0$, 且 $\delta_{2^n}^\alpha$ 是在 0 时刻从 $\delta_{2^n}^i$ 出发, 而在 k 时刻可到达的, 即 $(M_P^{(k)})_{\alpha,i} > 0$. 因此,

$$(M_P^{(k+1)})_{j,i} = \sum_{\beta=1}^{2^n} {}_{\mathcal{B}}(M_P)_{j,\beta}(M_P^{(k)})_{\beta,i} \geqslant (M_P)_{j,\alpha}(M_P^{(k)})_{\alpha,i} = 1 > 0.$$

并且, 如果 $\delta_{2^n}^j$ 是从 $\delta_{2^n}^i$ 可到达的, 则必有一条路径, 长度不超过 2^n, 它从 $\delta_{2^n}^i$ 出发, 到 $\delta_{2^n}^j$ 终结, 因为这里一共只有 2^n 个不同状态. 因此我们无须考虑 $s > 2^n$ 时的 $M_P^{(s)}$. 于是充分性获证.

(必要性) 假定存在 $\mathcal{R}_P$ 的一个元素 $r_{j,i} = 0$, 根据充分性的证明不难看出, $\delta_{2^n}^i$ 永远不可能到达 $\delta_{2^n}^j$. □

要证明联合可达与能控性的等价性, 我们需要以下两个引理, 它们可在任何关于随机过程的经典教科书找到, 例如文献 [16].

设 $\xi_n, n = 0, 1, 2, \cdots$ 为一齐次马尔可夫链, 记

$$p_n(j|i) = \mathbb{P}(\xi_n = j|\xi_0 = i),$$

$$F_{ji} = \mathbb{P}(\text{存在某个 } n \geqslant 1 \text{使得 } \xi_n = j \mid \xi_0 = i).$$

引理 11.3.1 (i) j 是常返的, 当且仅当,

$$\sum_{n=0}^{\infty} p_n(j|j) = \infty.$$

(ii) j 是过渡的, 当且仅当,

$$\sum_{n=0}^{\infty} p_n(j|i) < \infty, \quad \forall i.$$

引理 11.3.2 如果 $i \leftrightarrow j$ 且 j 是常返的, 那么 i 也是常返的, 并且,

$$F_{ii} = F_{ij} = F_{ji} = F_{jj} = 1.$$

现在可以证明如下定理了, 它表明联合可达与能控性等价.

定理 11.3.2 考察概率布尔控制网络 (11.2.11). 设激活模型 Σ_λ 的概率 $P_\lambda > 0, \forall \lambda \in \Lambda$, 那么, (11.2.11) 能控, 当且仅当,

$$\mathcal{R}_P > 0.$$

证明 必要性是显然的, 它的证明与定理 11.3.1 必要性的证明是一样的. 我们通过构造一组特殊的控制来证明充分性. 选择 $u(t)$ 在任何时刻 t 均为一组均匀分布, 则在 (11.2.14) 中, $L(t)u(t)$ 构成一随机逻辑矩阵. 对 (11.2.14) 两边取期望, 则得

$$\mathbb{E}(x(t+1)) = \mathbb{E}(L)\mathbb{E}(u)\mathbb{E}(x(t)),$$

这里,

$$\mathbb{E}(u) = [\underbrace{1/2^m, 1/2^m, \cdots, 1/2^m}_{2^m}]^{\mathrm{T}}.$$

因此, $x(t)$ 成为一个马尔可夫链, 其转移矩阵为 $\mathbb{E}(L)\mathbb{E}(u)$.

下面我们证明: 对任何两个状态 $\delta_{2^n}^i$, $\delta_{2^n}^j \in \Delta^n$, 简记为 $i, j \in \Delta^n$, j 与 i 连通. 由于该网络是联合可达的, 则存在 $\{(\Sigma_{\lambda_t}, u_{\lambda_t}), t = 0, \cdots, s-1\}$ 使得 j 从 i 可达. 因此,

$$\begin{aligned} p_s(j|i) &\geqslant \mathbb{P}(\text{在时间}t\text{选 } \Sigma_{\lambda_t} \text{以及 } u_{\lambda_t}, t = 0, \cdots, s-1) \\ &= \frac{1}{2^{ms}} \prod_{t=0}^{s-1} P_{\lambda_t} > 0. \end{aligned}$$

这就证明了 $i \to j$. 同理可证 $j \to i$, 故 $i \leftrightarrow j$.

其次, 我们证明: 至少存在一个常返态 $i^* \in \mathcal{D}^n$. 如其不然, 则由引理 11.3.1 可知

$$\sum_{s=0}^{\infty} p_s(j|i) < \infty, \quad i, j = 1, \cdots, 2^n.$$

于是,

$$\sum_{j=1}^{2^n} \sum_{s=0}^{\infty} p_s(j|i) < \infty,$$

因为它是有限个有限数的和. 但另一方面,

$$\sum_{j=1}^{2^n} \sum_{s=0}^{\infty} p_s(j|i) = \sum_{s=0}^{\infty} \sum_{j=1}^{2^n} p_s(j|i) = \sum_{s=0}^{\infty} 1 = \infty.$$

故得矛盾. 因此, 至少有一个 i^* 是常返的.

因为 i^* 是常返的, 则所有的点对都是连通的. 根据引理 11.3.2 则得

$$F_{ji} = 1, \quad i, j = 1, 2, \cdots, 2^n.$$

这说明: 当控制 $u(t)$ 以均匀分布随机选择时, 对任意 i 和 j, j 依概率 1 从 i 可达. 因此, 该网络 (依概率 1) 能控. □

注 11.3.1 在定理 11.3.2 的证明中, 控制是按均匀分布随机选的. 实际上, 按任何分布选择控制都可以, 只要每一个控制值被取到的概率都大于零即可.

下面给一个简单例子.

例 11.3.1 回忆例 11.2.2. 利用各模态的结构矩阵 (11.2.16), 以及式 (11.3.2) 和 (11.3.3) 可得

$$M_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \quad M_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad M_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix},$$

$$M_4 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad M_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \quad M_6 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

并且,

$$M_P = \sum_{\lambda=1}^{6} {}_{\mathcal{B}} M_\lambda = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}.$$

不难算出

$$M_P^{(2)} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix},$$

于是有 $\mathcal{R}_P = \sum_{s=1}^{4} M_P^{(s)} > 0$. 故网络 (11.2.16) 是能控的.

11.4　稳定性与镇定

11.4.1　概率布尔网络的稳定性

定义 11.4.1　考察概率布尔网络 (11.2.1). (11.2.1) 称为 (依概率 1) 全局稳定的, 如果存在 X_e 使对任何 X_0,

$$\mathbb{P}\left(\lim_{t\to\infty} X(t) = X_e \,\middle|\, X(0) = X_0\right) = 1. \tag{11.4.1}$$

下面这个引理不仅在后面常用, 而且自身也是有趣的.

引理 11.4.1　考察概率布尔网络 (11.2.1), 设 $P_\lambda > 0$, $\lambda = 1, 2, \cdots, N$, 记

$$A_\lambda^T = \{\Sigma_\lambda \text{连续出现 } T\text{次}\}.$$

那么, 对任何 $T > 0$,

$$\mathbb{P}(A_\lambda^T\text{出现无穷次}) = 1.$$

证明　首先证明 $\mathbb{P}(A_\lambda^T) = 1$. 对任何整数 $k \geqslant 0$, 我们有

$$\mathbb{P}(\Sigma(t) = \Sigma_\lambda, k \leqslant t < k + T) = P_\lambda^T > 0.$$

因此,

$$\begin{aligned}\mathbb{P}(A_\lambda^T) &\geqslant \mathbb{P}(\exists m, \forall t, mT \leqslant t < (m+1)T, \Sigma(t) = \Sigma_\lambda)\\ &= 1 - \mathbb{P}(\forall m, \exists t, mT \leqslant t < (m+1)T, \Sigma(t) \neq \Sigma_\lambda)\\ &= 1 - \lim_{m\to\infty}(1 - P_\lambda^T)^m = 1.\end{aligned}$$

记 $B_m = \{A_\lambda^T\text{至少发生 } m\text{次}\}$, 不难看出 $B_1, B_2, \cdots$ 是事件的一个收缩集, 即 $B_1 \supset B_2 \supset \cdots$. 记 $B = \bigcap_{m=1}^{\infty} B_m = \{A_\lambda^T\text{发生无穷次}\}$, 那么,

$$\mathbb{P}(A_\lambda^T\text{发生无穷次}) = \lim_{m\to\infty} \mathbb{P}(B_m).$$

因为模态 $\Sigma(t)$, $t = 0, 1, 2, \cdots$ 是独立的, 并且, 如果 A_λ^T 发生至少 $m+1$ 次, 它必需首次后再连续发生至少 m 次. 因此,

$$\begin{aligned}\mathbb{P}(B_{m+1}) &= \sum_{a=0}^{\infty} \mathbb{P}(\text{在时间 } a \leqslant t < a + T\text{首次发生 } A_\lambda^T)\mathbb{P}(B_m)\\ &= \mathbb{P}(B_1)\mathbb{P}(B_m) = \mathbb{P}(A_\lambda^T)\mathbb{P}(B_m) = \mathbb{P}(B_m).\end{aligned}$$

于是可得结论: $\mathbb{P}(A_\lambda^T\text{至少发生无穷次 }) = 1$. □

引理 11.4.2　考察概率布尔网络 (11.2.1), 设 $P_\lambda > 0$, $\lambda = 1, 2, \cdots, N$. 如果它 (依概率 1) 全局收敛于 X_e, 那么, x_e 是所有模态 $\Sigma_\lambda, \lambda = 1, 2, \cdots, N$ 的不动点.

证明　假定 X_e 不是 Σ_λ 的不动点, 由引理 11.4.1 可知, 对任何 $T > 0$, A_λ^T 将依概率 1 出现无穷多次. 一旦 A_λ^T 出现, 则必有一个状态在这段连续时间 T 里不等于 x_e, 因为 x_e 不是这个模态的不动点. 因此, 当 A_λ^T 发生无数次时, 系统不可能收敛到 X_e. 即, 系统收敛到 X_e 的概率为 0. □

为了提供充要条件, 我们还需要以下两个关于马尔可夫链性质的引理, 它们均可见于文献 [16].

引理 11.4.3　给定一个马尔可夫链 ξ_n. 状态 j 是常返的, 当且仅当,

$$\mathbb{P}(\text{有无数个 } n \text{使得 } \xi_n = j \mid \xi_0 = j) = 1, \tag{11.4.2}$$

状态 j 是过渡的, 当且仅当,

$$\mathbb{P}(\text{有无数个 } n \text{使得 } \xi_n = j \mid \xi_0 = i) = 0, \quad \forall i. \tag{11.4.3}$$

引理 11.4.4　给定一个马尔可夫链 ξ_n. 如果 $i \to j$ 并且 i 是常返的, 那么 $i \leftrightarrow j$ 并且 j 也是常返的.

下面的定理给出稳定的充要条件.

定理 11.4.1　考察概率布尔网络 (11.2.1), 设 $P_\lambda > 0$, $\lambda = 1, 2, \cdots, N$. 令

$$L = \sum_{\lambda=1}^{N} {}_{\mathcal{B}} L_\lambda,$$

这里, L_λ 是 Σ_λ 的结构矩阵. 那么, 该网络 (依概率 1) 全局稳定于 $x_e = \delta_{2^n}^i$, 当且仅当, x_e 是所有模态 $\Sigma_\lambda, \lambda = 1, 2, \cdots, N$ 的不动点, 并且,

$$\mathrm{Row}_i \left(\sum_{s=1}^{2^n} {}_{\mathcal{B}} L^{(s)} \right) > 0. \tag{11.4.4}$$

证明　(必要性) 由定理 11.3.1 的证明可以看出, (11.4.4) 表明, 存在一个模态序列 $\Sigma_{\lambda_0}, \cdots, \Sigma_{\lambda_{s-1}}$, 使得 $x_e = \delta_{2^n}^i$ 在时刻 s 可达, 只要在任何 $t < s$ 时刻, 激活的模态为 Σ_{λ_t}. 这显然对稳定是必要的.

(充分性) 与定理 11.3.1 的证明类似, 可知, 如果 (11.4.4) 成立, 那么, 对任意 $\delta_{2^n}^j, j = 1, \cdots, 2^n$, 存在一个 $s_j \leqslant 2^n$, 使得

$$p_{s_j}(i|j) := \mathbb{P}(x(s_j) = \delta_{2^n}^i | x(0) = \delta_{2^n}^j) > 0,$$

即 $j \to i$. 同时我们知道, 由于一个概率布尔网络只有有限个状态, 因此至少有一个状态是常返的. 我们证明 $x_e = \delta_{2^n}^i$ 是唯一的常返态. 否则假定存在一个 $j \neq i$, 使 $\delta_{2^n}^j$ 是另一个常返态. 由于 $j \to i$, 根据引理 11.4.4 可知 $i \leftrightarrow j$. 但是 $x_e = \delta_{2^n}^i$ 是所有模态的不动点, 它的可达集只含自己, 因此, $i \leftrightarrow j$ 是不可能的. 由此可知, 只有 x_e 是常返的, 而其他状态都是过渡的. 由引理 11.4.3 可知, 其他状态依概率 1 只可达有限次, 因此轨道依概率 1 收敛于 X_e. □

例 11.4.1 考察概率布尔网络

$$\begin{cases} X_1(t+1) = F^1(X_1(t), X_2(t)), \\ X_2(t+1) = F^2(X_1(t), X_2(t)), \end{cases} \tag{11.4.5}$$

这里 $F^1 \in \{F_1^1,\ F_2^1,\ F_3^1\}$, $F^2 \in \{F_1^2,\ F_2^2,\ F_3^2\}$,

$$\begin{aligned} &F_1^1(X_1, X_2) = X_1 \vee X_2, && F_2^1(X_1, X_2) = \neg(X_1 \to X_2), \\ &F_3^1(X_1, X_2) = X_1 \bar{\vee} X_2, && F_1^2(X_1, X_2) = X_1 \wedge X_2, \\ &F_2^2(X_1, X_2) = \neg(X_1 \vee X_2), && F_3^2(X_1, X_2) = X_1 \to X_2. \end{aligned}$$

于是, 系统共有 9 个模态, 其结构矩阵分别为

$$\begin{aligned} &L_1 = \delta_4[1,\ 2,\ 2,\ 4], \quad L_2 = \delta_4[2,\ 2,\ 2,\ 3], \quad L_3 = \delta_4[1,\ 2,\ 1,\ 3], \\ &L_4 = \delta_4[3,\ 2,\ 4,\ 4], \quad L_5 = \delta_4[4,\ 2,\ 4,\ 3], \quad L_6 = \delta_4[3,\ 2,\ 3,\ 3], \\ &L_7 = \delta_4[3,\ 2,\ 2,\ 4], \quad L_8 = \delta_4[4,\ 2,\ 2,\ 3], \quad L_9 = \delta_4[3,\ 2,\ 1,\ 3]. \end{aligned} \tag{11.4.6}$$

不难验证, 各模态的唯一共同不动点是 $x_e = \delta_4^2$. 然后检验条件 (11.4.4):

$$L = \sum_{\lambda=1}^{9} {}_{\mathcal{B}} L_\lambda = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}.$$

于是有

$$L^{(2)} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}.$$

由于 $\mathrm{Row}_2(L^{(2)})$ 已经大于 0, 于是 (11.4.4) 成立. 根据定理 11.4.1, 网络 (11.4.5) 依概率 1 稳定于 δ_4^2.

11.4.2 概率布尔网络的镇定

定义 11.4.2 考察一个布尔控制网络, 设其代数状态空间表示为

$$x(t+1) = Lu(t)x(t).$$

一个点 X_e 称为受控不动点 , 如果存在控制 U_e, 使得

$$x_e = Lu_e x_e.$$

显然如下命题成立.

命题 11.4.1 $x_e = \delta_{2^n}^i$ 是关于控制 $u_e = \delta_{2^m}^j$ 的受控不动点, 当且仅当

$$x_e = \mathrm{Col}_i(\mathrm{Blk}_j(L)).$$

定义 11.4.3 考察概率布尔网络 (11.2.1), 设 P_λ 为 Σ_i 被激活的概率, $\lambda = 1, 2, \cdots, N$. 称网络 (11.2.1) 可 (依概率 1) 被镇定到 X_e, 如果存在一个控制序列, 使受控系统依概率 1 收敛到 $X_e \in \mathcal{D}^n$.

由概率布尔网络稳定性条件不难想到以下的引理.

引理 11.4.5 考察概率布尔网络 (11.2.1), 设 $P_\lambda > 0$, $\lambda = 1, 2, \cdots, N$. 该网络若要镇定到 X_e, 则 X_e 必须是所有模态共同的受控不动点.

证明 类似于引理 11.4.2 的证明即可知, X_e 必须是一个共同的受控不动点. □

下面的定理给出可镇定的充要条件.

定理 11.4.2 考察概率布尔网络 (11.2.1), 设 $P_\lambda > 0$, $\lambda = 1, 2, \cdots, N$. 该网络可镇定到 $x_e = \delta_{2^n}^i$, 当且仅当,

(i) X_e 是所有模态共同的受控不动点;

(ii)
$$\mathrm{Row}_i(\mathcal{R}_P) > 0. \tag{11.4.7}$$

证明 必要性是显然的, 我们证明充分性. 类似于定理 11.3.2 的证明, 我们依均匀分布随机地选择控制 $u(t)$. 于是 $x(t)$ 成为一个马尔可夫链, 且其转移矩阵为 $\mathbb{E}(L)\mathbb{E}(u)$. 简记 $j \sim \delta_{2^n}^j$. 根据定理 11.3.1 可知 (11.4.7) 成立, 于是对任何状态 $j = 1, 2, \cdots, 2^n$, 均有 $j \to i$. 下面将状态分为两类:

$$T = \{j \in \mathcal{D}^n | j \to i, i \nrightarrow j\},$$

$$R = \{j \in \mathcal{D}^n | j \leftrightarrow i\}.$$

我们断言: T 中的状态都是过渡的, R 中的状态都是常返的.

首先因为状态空间是有限集, 故必然存在常返常返态 j^*. 由于 $j^* \to i$, 根据引理 11.4.4, $j^* \leftrightarrow i$ 并且 i 是常返的. 再根据引理 11.3.2 可知, R 中所有状态均常返.

再考虑 T, 设 $j \in T$ 常返, 则得 $j \leftrightarrow i$, 矛盾. 因此, $j \in T$ 都是过渡的.

现在, 根据引理 11.4.3 可知, 所有 T 中状态依概率 1 只能到达有限次, 因此, 从任何状态出发, 轨线最终会依概率 1 进入 R. 再根据引理 11.3.2, 它依概率 1 到达 $x_e = i$. 一旦到达 x_e, 则可固定控制为 u_e. 因为 x_e 是在 u_e 下的共同受控不动点, 轨线此后将终止于 x_e. □

注 11.4.1 如果模态 $\Sigma(t) \in \{\Sigma_\lambda, \lambda = 1, 2, \cdots, N\}$ 是可观测的, 那么, 可以将引理 11.4.5 中统一的 u_e 用不同的 $u_e^\lambda, \lambda = 1, 2, \cdots, N$ 来代替, 引理仍成立. 同时, 定理 11.4.2 在该替代下也仍成立.

下面给出一个例子.

例 11.4.2 回顾例 11.2.2.

(i) 利用结构矩阵 (11.2.17) 并根据命题 11.4.1 可知:

δ_4^1 是模态 $\Sigma_3, \Sigma_4, \Sigma_5, \Sigma_6$ 在控制 δ_2^1 下的不动点, 也是模态 $\Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4$ 在控制 δ_2^2 下的不动点.

δ_4^2 是模态 $\Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4$ 在控制 δ_2^1 下的不动点.

δ_4^3 是模态 Σ_2 在控制 δ_2^1 下的不动点, 也是模态 Σ_6 在控制 δ_2^2 下的不动点.

δ_4^4 是模态 Σ_5 在控制 δ_2^1 下的不动点, 也是模态 $\Sigma_1, \Sigma_3, \Sigma_5$ 在控制 δ_2^2 下的不动点.

因此, 不存在统一的控制下的受控不动点, 因此由引理 11.4.5 可知, 网络 (11.2.14) 不能镇定.

(ii) 由于 δ_4^1 是在依赖于模型的控制下的共同受控不动点, 如果 $\Sigma(t)$ 是可观测的, 则根据注 11.4.1 可知, 网络 (11.2.14) 能镇定.

(iii) 设网络 (11.2.14) 中的 $F_2 = F_2^1$ 不变, 则 (11.2.14) 只有三个不同模态: Σ_1, Σ_3, Σ_5. 因此, δ_4^4 是所有模态在控制 δ_2^2 下的共同受控不动点. 检验条件 (11.4.7):

$$M'_P = M_1 +_{\mathcal{B}} M_3 +_{\mathcal{B}} M_5 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

直接计算可得

$$\mathcal{R}'_P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

因为 $\mathrm{Row}_4(\mathcal{R}_P) > 0$, 可知现在网络 (11.2.14) 可镇定到 δ_4^4.

此外, 由 $\mathcal{R}_P$ 可知, 当控制随机选取时

$$T = \{\delta_4^1, \delta_4^3\}, \quad R = \{\delta_4^2, \delta_4^4\}.$$

11.5　概率布尔网络的能观性

本节讨论概率布尔网络的能观性, 解决的方法是利用集合能控性, 其基本思路与第 5 章中的辅助系统方法是一样的. 本节主要内容来自文献 [151], 记号也尽量与文献 [151] 保持一致.

11.5.1　模型与能观性

考察概率布尔网络

$$\begin{cases} X_1(t+1) = F^1(X_1(t), X_2(t), \cdots, X_n(t)), \\ X_2(t+1) = F^2(X_1(t), X_2(t), \cdots, X_n(t)), \\ \qquad\qquad \cdots\cdots \\ X_n(t+1) = F^n(X_1(t), X_2(t), \cdots, X_n(t)), \end{cases}$$
$$Y_j(t) = C_j(X_1(t), X_2(t), \cdots, X_n(t)), \quad j = 1, 2, \cdots, s. \tag{11.5.1}$$

记其代数状态空间表示为

$$\begin{cases} x(t+1) = L(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{11.5.2}$$

这里

$$L(t) \in \{L_1, L_2, \cdots, L_N\} \subset \mathcal{L}_{2^n \times 2^n}, \quad t = 0, 1, \cdots$$

为一独立同分布逻辑矩阵序列, 满足

$$\mathbb{P}(L(t) = L_i) = p_i, \quad i = 1, 2, \cdots, N, \tag{11.5.3}$$

其中

$$\sum_{i=1}^{N} p_i = 1.$$

记

$$\mathbf{P}^X = \sum_{\lambda=1}^{N} p_\lambda L_\lambda. \tag{11.5.4}$$

显然, $X(t)$ 是一个马尔可夫链, $\mathbf{P}^X$ 为其概率转移矩阵. 记

$$\mathbf{P}^X(k) = \left(\mathbf{P}^X\right)^k,$$

则 $\mathbf{P}^X(k)$ 为 $X(t)$ 的 k 步转移矩阵, 即 $\left[\mathbf{P}^X(k)\right]_{i,j}$ 为状态经 k 步从 $\delta_{2^n}^j$ 转移到 $\delta_{2^n}^i$ 的概率.

记

$$\mathbf{L} := (L_1, L_2, \cdots, L_N).$$

定义序列 $w(t) \in \Delta_N$, $t = 0, 1, \cdots$ 如下:

$$w(t) = \delta_N^i \Leftrightarrow L(t) = L_i, \quad t = 0, 1, 2, \cdots.$$

那么, (11.5.2) 可改写为

$$\begin{cases} x(t+1) = \mathbf{L}w(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{11.5.5}$$

给定 w, 则决定了一族独立同分布逻辑矩阵序列 $(L(0), L(1), L(2), \cdots)$, 记 $\mathbf{X}(T, w, X_0)$ 为从 X_0 出发依 w 生成的一条长度为 $T+1$ 的轨线, 即

$$\mathbf{X}(T, w, X_0) = (X_0, X(1, w, X_0), X(2, w, X_0), \cdots, X(T, w, X_0)).$$

其相应的输出轨线为

$$\mathbf{Y}(T, w, X_0) = (Y_0 = h(X_0), Y(1, w, X_0), Y(2, w, X_0), \cdots, Y(T, w, X_0)).$$

下面介绍三种概率布尔网络的可观性.

定义 11.5.1 [141] 概率布尔网络 (11.5.1) 称为在有限时间 $[0, T]$ 依概率可观, 如果对两个初始点 $X_0 \neq \bar{X}_0$, 有

$$\mathbb{P}\left(\mathbf{Y}(T, w, X_0) \neq \mathbf{Y}(T, w, \bar{X}_0)\right) > 0. \tag{11.5.6}$$

定义 11.5.2 [151] 概率布尔网络 (11.5.1) 称为在有限时间 $[0, T]$ 依概率 1 可观, 如果对两个初始点 $X_0 \neq \bar{X}_0$, 有

$$\mathbb{P}\left(\mathbf{Y}(T, w, X_0) \neq \mathbf{Y}(T, w, \bar{X}_0)\right) = 1. \tag{11.5.7}$$

定义 11.5.3 [151] 概率布尔网络 (11.5.1) 称为依分布渐近可观, 如果对两个初始点 $X_0 \neq \bar{X}_0$, 有

$$\lim_{t\to\infty} \mathbb{P}\left(\mathbf{Y}(t, w, X_0) \neq \mathbf{Y}(t, w, \bar{X}_0)\right) = 1. \tag{11.5.8}$$

11.5.2 集合能达性

定义 11.5.4 [151] 考察概率布尔网络 (11.5.1). 设 $\mathcal{M}_0$, $\mathcal{M}_d \subset \Delta_{2^n}$ 分别为初始集和目标集, 称网络从 $\mathcal{M}_0$ 到 $\mathcal{M}_d$ 在时间 $[0,T]$ 内依概率 1 可达, 如果对任一 $X_0 \in \mathcal{M}_0$

$$\mathbb{P}\left(\exists 0 \leqslant k \leqslant T, \text{使得 } X(k,w,X_0) \in \mathcal{M}_d\right) = 1. \tag{11.5.9}$$

定义 11.5.5 [151] 考察概率布尔网络 (11.5.1). 设 $\mathcal{M}_0$, $\mathcal{M}_d \subset \Delta_{2^n}$ 分别为初始集和目标集, 称网络从 $\mathcal{M}_0$ 到 $\mathcal{M}_d$ 依分布渐近可达, 如果对任一 $X_0 \in \mathcal{M}_0$,

$$\lim_{t\to\infty} \mathbb{P}\left(\exists 0 \leqslant k \leqslant t, \text{使得 } X(k,w,X_0) \in \mathcal{M}_d\right) = 1. \tag{11.5.10}$$

注意, (11.5.10) 与下式等价:

$$\mathbb{P}\Big(\exists 0 \leqslant k < \infty, \text{使得 } X(k,w,X_0) \in \mathcal{M}_d\Big) = 1. \tag{11.5.11}$$

设 $\mathcal{M} \subset \Delta_{2^n}$ 为一状态子集, 定义其示性矩阵 $\mathbf{D}_{\mathcal{M}} \in \mathcal{B}_{2^n\times 2^n}$ 如下:

$$\mathrm{Col}_j(\mathbf{D}_{\mathcal{M}}) := \begin{cases} \delta_{2^n}^j, & \delta_{2^n}^j \in \mathcal{M}, \\ \delta_{2^n}^0, & \delta_{2^n}^j \notin \mathcal{M}, \end{cases} \tag{11.5.12}$$

这里 $\delta_{2^n}^0 = \mathbf{0}_{2^n}$.

将状态空间扩展为

$$\Delta_{2^n}^0 := \Delta_{2^n} \cup \{\delta_{2^n}^0\}.$$

在 $\Delta_{2^n}^0$ 上定义一个逻辑动态系统

$$\begin{aligned} \tilde{x}(t+1) &= \mathbf{L}w(t)\mathbf{D}_{\mathcal{M}_d^c}\tilde{x}(t) \\ &= \mathbf{L}W_{[2^n,N]}\mathbf{D}_{\mathcal{M}_d^c}W_{[N,2^n]}\tilde{x}(t), \end{aligned} \tag{11.5.13}$$

这里① $\mathcal{M}_d^c = \Delta_{2^n} \backslash \mathcal{M}_d$ 是 $\mathcal{M}_d$ 的余集.

---

① 如果 $A \in \mathcal{M}_{m\times m}$, 且 $B \in \mathcal{M}_{n\times n}$, 那么,

$$W_{[m,n]}(A \otimes B)W_{[n,m]} = B \otimes A.$$

设 $D \in \mathcal{M}_{n\times n}$, 将上述公式用于 $(I_m \otimes D)$ 可得

$$I_m \otimes D = W_{[n,m]}(D \otimes I_m)W_{[m,n]}.$$

于是, 设 $X \in \mathbb{R}^m$, $D \in \mathcal{M}_{n\times n}$, 则

$$XD = (I_m \otimes D)X = W_{[n,m]}DW_{[m,n]}X.$$

式 (11.5.13) 使用了这个等式.

记

$$\left[\mathbf{P}^{\tilde{X}}\right]_{i,j} := \mathbb{P}\left(\tilde{X}(t+1) = \delta_{2^n}^i \;\middle|\; \tilde{X}(t) = \delta_{2^n}^j\right).$$

由 (11.5.13) 可知

$$\mathbf{P}^{\tilde{X}} = \mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}. \tag{11.5.14}$$

因此,

$$\left[\mathbf{P}^{\tilde{X}}(k)\right]_{i,j} = \mathbb{P}\left(\tilde{X}(t+k) = \delta_{2^n}^i \;\middle|\; \tilde{X}(t) = \delta_{2^n}^j\right), \tag{11.5.15}$$

这里,

$$\mathbf{P}^{\tilde{X}}(k) = \left[\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right]^k.$$

实际上, $\mathbf{P}^{\tilde{X}}$ 中的非零元素表示非零状态之间的转移概率, 于是可以得到如下关系式, 用以计算从非零状态到零状态的概率.

引理 11.5.1

$$\mathbb{P}\left(\tilde{x}(t+k) = \delta_{2^n}^0 \mid \tilde{x}(t) = \delta_{2^n}^j\right) = 1 - \sum_{i=0}^{2^n}\left[\left(\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right)^k\right]_{i,j}, \quad 1 \leqslant j \leqslant 2^n. \tag{11.5.16}$$

根据引理 11.5.1 可知, 逻辑动态系统 (11.5.13) 的一步转移概率为

$$\begin{aligned}
\tilde{p}_{i,j} &:= \left[\mathbf{P}^{\tilde{X}}\right]_{i,j} = \mathbb{P}\left(\tilde{x}(t+1) = \delta_{2^n}^i \mid \tilde{x}(t) = \delta_{2^n}^j\right) \\
&= \begin{cases} \left[\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right]_{i,j}, & 1 \leqslant i, j \leqslant 2^n, \\ 0, & i = 0, j \in \Lambda_d^c, \\ 1, & i = 0, j \in \{0\} \cup \Lambda_d, \\ 0, & i \neq 0, j = 0. \end{cases}
\end{aligned} \tag{11.5.17}$$

这里, $j \in \Lambda_d \Leftrightarrow \delta_{2^n}^j \in \mathcal{M}_d$, $j \in \Lambda_d^c \Leftrightarrow \delta_{2^n}^j \in \mathcal{M}_d^c$.

例 11.5.1 考察概率布尔网络 (11.5.5), 设其 (一步) 概率转移矩阵为

$$\mathbf{P}^X = \begin{bmatrix}
0 & 0 & 0.1 & 0 & 0 & 0 & 0.2 & 0 \\
0.7 & 0.6 & 0.3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.7 & 0 & 0 & 0 & 0 \\
0 & 0.4 & 0 & 0.3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.6 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.7 & 1 & 0 & 0 \\
0.3 & 0 & 0 & 0 & 0 & 0 & 0.8 & 1
\end{bmatrix}.$$

设 $\mathcal{M}_d=\delta_8\{2,5,8\}$, 那么, $\mathcal{M}_d^c$ 的示性矩阵为

$$\mathbf{D}_{\mathcal{M}_d^c}=\delta_8[1,0,1,1,0,1,1,0].$$

于是

$$\mathbf{P}^{\tilde{X}}=\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}=\begin{bmatrix}0&0&0.1&0&0&0&0.2&0\\0.7&0&0.3&0&0&0&0&0\\0&0&0&0.7&0&0&0&0\\0&0&0&0.3&0&0&0&0\\0&0&0.6&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0\\0.3&0&0&0&0&0&0.8&0\end{bmatrix}.$$

引理 11.5.2 设 $\mathcal{M}_0\subset\Delta_{2^n}$ 为初始集, $\mathcal{M}_d\subset\Delta_{2^n}$ 为目标集.

(i) 概率布尔网络 (11.5.5) 从 $\mathcal{M}_0$ 在 $0\leqslant t\leqslant T$ 可达 $\mathcal{M}_d$, 当且仅当, 概率逻辑网络 (11.5.13) 从 $\mathcal{M}_0$ 在 $1\leqslant t\leqslant T+1$ 可达 $\delta_{2^n}^0$.

(ii) 概率布尔网络 (11.5.5) 从 $\mathcal{M}_0$ 依分布渐近可达 $\mathcal{M}_d$, 当且仅当, 概率逻辑网络 (11.5.13) 从 $\mathcal{M}_0$ 依分布渐近可达 $\delta_{2^n}^0$.

证明 设 $x_0\in\mathcal{M}_0$ 为初态, 且 $\tilde{x}(t,w,x_0)$ 为 (11.5.13) 的一条轨线, 那么,

$$\begin{aligned}&\mathbb{P}\left(\exists 1\leqslant k\leqslant t+1, \text{使得 } x(k,w,x_0)\in\mathcal{M}_d\right)\\=&\mathbb{P}\left(\{x_0\in\mathcal{M}_d\}\cup\{x(1,w,x_0)\in\mathcal{M}_d\}\cup\cdots\cup\{x(k,w,x_0)\in\mathcal{M}_d\}\right)\\=&\mathbb{P}\left(\bigcup_{j=1}^{t+1}\{\tilde{x}(j,w,x_0)=\delta_{2^n}^0\}\right)\\=&\mathbb{P}\left(\exists 1\leqslant k\leqslant t+1, \text{使得 } \tilde{x}(k,w,x_0)=\delta_{2^n}^0\right).\end{aligned}\tag{11.5.18}$$

利用 (11.5.18) 不难看出 (11.5.9) 等价于

$$\mathbb{P}\left(\exists 1\leqslant k\leqslant T+1, \text{使得 } \tilde{x}(k,w,x_0)=\delta_{2^n}^0\right)=1;\tag{11.5.19}$$

且 (11.5.10) 等价于

$$\lim_{t\to\infty}\mathbb{P}\left(\exists 1\leqslant k\leqslant t+1, \text{使得 } \tilde{x}(k,w,x_0)=\delta_{2^n}^0\right)=1.\tag{11.5.20}$$

于是, 引理 11.5.2 的两个断言成立. □

下面这个引理是显然的.

引理 11.5.3 考察概率逻辑网络 (11.5.13), 从任何状态 k 步到达 $\delta_{2^n}^0$ 的概率是单调增的, 即

$$\mathbb{P}\left(\tilde{x}(k+1)=\delta_{2^n}^0\mid\tilde{x}(0)=\delta_{2^n}^j\right)\geqslant\mathbb{P}\left(\tilde{x}(k)=\delta_{2^n}^0\mid\tilde{x}(0)=\delta_{2^n}^j\right).\tag{11.5.21}$$

引理 11.5.4　考察概率逻辑网络 (11.5.13).

(i) 从 $\mathcal{M}_0$ 出发, 可以在时间 $1 \leqslant t \leqslant T+1$ 到达零状态 $\delta_{2^n}^0$, 当且仅当, 对任何 $x_0 \in \mathcal{M}_0$

$$\mathbb{P}\left(\tilde{x}(t,w,x_0) = \delta_{2^n}^0\right) = 1, \quad t \geqslant T+1. \tag{11.5.22}$$

(ii) 从 $\mathcal{M}_0$ 出发, 状态依分布渐近可达 $\delta_{2^n}^0$, 当且仅当, 从任何 $x_0 \in \mathcal{M}_0$ 出发的轨线依分布收敛于 $\delta_{2^n}^0$, 即

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t,w,x_0) = \delta_{2^n}^0\right) = 1, \quad \forall x_0 \in \mathcal{M}_0. \tag{11.5.23}$$

证明　首先证明下面这个断言: 对任何 $x_0 \in \mathcal{M}_0$ 和任何 $T \geqslant 1$ 均有

$$\mathbb{P}\left(\tilde{x}(T,w,x_0) = \delta_{2^n}^0\right) = \mathbb{P}\left(\exists 1 \leqslant k \leqslant T, \text{使得 } \tilde{x}(k,w,x_0) = \delta_{2^n}^0\right). \tag{11.5.24}$$

利用数学归纳法, 显然 (11.5.24) 对 $T=1$ 成立. 设其对 $T=s$ 成立, 则当 $T=s+1$ 时有

$$\begin{aligned}
&\mathbb{P}\left(\tilde{x}(s+1,w,x_0) = \delta_{2^n}^0\right) \\
&= \sum_{x_s \in \Delta_{2^n}^0} \mathbb{P}\left(\tilde{x}(s,w,x_0) = x_s\right) \cdot \mathbb{P}\left(\tilde{x}(1,w,x_s) = \delta_{2^n}^0\right) \\
&= \sum_{x_s \in \Delta_{2^n}} \mathbb{P}\left(\tilde{x}(s,w,x_0) = x_s\right) \cdot \mathbb{P}\left(\tilde{x}(1,w,x_s) = \delta_{2^n}^0\right) \\
&\quad + \mathbb{P}\left(\tilde{x}(s,w,x_0) = \delta_{2^n}^0\right) \\
&= \mathbb{P}\left(\tilde{x}(k) \neq \delta_{2^n}^0, 1 \leqslant k \leqslant s,\ \tilde{x}(s+1) = \delta_{2^n}^0 \mid \tilde{x}_0 = x_0\right) \\
&\quad + \mathbb{P}\left(\exists 1 \leqslant k \leqslant s, \text{使得 } \tilde{x}(k,w,x_0) = \delta_{2^n}^0\right) \\
&= \mathbb{P}\left(\exists 1 \leqslant k \leqslant s+1, \text{使得 } \tilde{x}(k,w,x_0) = \delta_{2^n}^0\right).
\end{aligned}$$

这就证明了 (11.5.24).

根据定义 11.5.4 及 11.5.5, 由 (11.5.24) 及引理 11.5.3 即可得 (11.5.22) 与 (11.5.23). □

由引理 11.5.2 及引理 11.5.4 立得以下结论.

命题 11.5.1　设初始集 $\mathcal{M}_0$ 及目标集 $\mathcal{M}_d$ 给定.

(i) 网络 (11.5.5) 从 $\mathcal{M}_0$ 出发可依概率 1 在有限时间 $0 \leqslant t \leqslant T$ 内到达 $\mathcal{M}_d$, 当且仅当, 网络 (11.5.13) 从任何 $x_0 \in \mathcal{M}_0$ 出发的轨线可依概率 1 在有限时间 $1 \leqslant t \leqslant T+1$ 内收敛于 $\delta_{2^n}^0$.

(ii) 网络 (11.5.5) 从 $\mathcal{M}_0$ 出发依分布渐近可达 $\mathcal{M}_d$, 当且仅当, 网络 (11.5.13) 从 $\mathcal{M}_0$ 出发依分布渐近收敛于 $\delta_{2^n}^0$.

11.5.3　依概率 1 有限时间可达

引理 11.5.5　考察网络 (11.5.13), 如果有一容许轨道从 x_0 到 $\mathcal{M}_d$, 则有一长度 $\ell \leqslant |\mathcal{M}_d^c|$ 的容许轨道从 x_0 到 $\mathcal{M}_d$, 等价地, 有一长度 $\ell \leqslant |\mathcal{M}_d^c|+1$ 的容许轨道从 x_0 到 $\delta_{2^n}^0$.

证明　不失一般性, 设 $x_0 \in \mathcal{M}_d^c$. 如果有一条容许轨道从 x_0 到 $\mathcal{M}_d$, 长度 $\ell > |\mathcal{M}_d^c|$, 则轨道中有相同的状态. 将这种 "环" 全部除去, 即得长度 $\ell \leqslant |\mathcal{M}_d^c|$ 的容许轨道. □

命题 11.5.2　考察网络 (11.5.5), 设

$$\mathcal{M}_0 = \left\{\delta_{2^n}^j \mid j \in \Lambda_0\right\},$$

这里 $\Lambda_0 \subset \{1,2,\cdots,2^n\}$.

(i) $\mathcal{M}_d$ 是从 $\mathcal{M}_0$ 出发, 在时间 $0 \leqslant t \leqslant T$ 内依概率 1 可达的, 当且仅当,

$$\mathrm{Col}_j\left[\left(\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right)^{T+1}\right] = \delta_{2^n}^0, \quad \forall j \in \Lambda_0. \tag{11.5.25}$$

(ii) $\mathcal{M}_d$ 是从 $\mathcal{M}_0$ 出发, 在有限时间内依概率 1 可达的, 当且仅当,

$$\mathrm{Col}_j\left[\left(\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right)^{|\mathcal{M}_d^c|+1}\right] = \delta_{2^n}^0, \quad \forall j \in \Lambda_0. \tag{11.5.26}$$

证明　注意到 $\delta_{2^n}^0$ 是网络 (11.5.13) 的一个平衡点, 根据命题 11.5.1, 对任何 $x_0 = \delta_{2^n}^j$ 及任何 $t \geqslant 0$,

$$\begin{aligned}
&\mathbb{P}\left(\exists 0 \leqslant k \leqslant t, \text{使得 } x(k,w,x_0) \in \mathcal{M}_d\right) \\
=& \mathbb{P}\left(\exists 1 \leqslant k \leqslant t+1, \text{使得 } \tilde{x}(k,w,x_0) = \delta_{2^n}^0\right) \\
=& \mathbb{P}\left(\tilde{x}(t+1,w,x_0) = \delta_{2^n}^0\right) \\
=& 1 - \sum_{i=1}^{2^n}\left[\left(\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right)^{t+1}\right]_{i,j}.
\end{aligned} \tag{11.5.27}$$

根据 (11.5.27) 以及命题 11.5.1 (i), $\mathcal{M}_d$ 是从 $\mathcal{M}_0$ 在时间 $0 \leqslant t \leqslant T$ 内依概率 1 可达, 当且仅当, 对某个 $0 \leqslant t \leqslant T$,

$$1 - \sum_{i=1}^{2^n}\left[\left(\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right)^{t+1}\right]_{i,j} = 1, \quad \forall j \in \Lambda_0.$$

也就是

$$\sum_{i=1}^{2^n}\left[\left(\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right)^{t+1}\right]_{i,j} = 0, \quad \forall j \in \Lambda_0. \tag{11.5.28}$$

根据引理 11.5.3, (11.5.28) 与 (11.5.25) 等价, 这就证明了结论 (i).

根据引理 11.5.5, $\mathcal{M}_d$ 是从 $\mathcal{M}_0$ 在有限时间内依概率 1 可达, 当且仅当, $\mathcal{M}_d$ 是从 $\mathcal{M}_0$ 在 $0 \leqslant t \leqslant |\mathcal{M}_d^c|$ 依概率 1 可达的. 于是, 结论 (ii) 就可从结论 (i) 推出. □

11.5.4 依分布渐近可达

定义 11.5.6 考察网络 (11.5.13).

(i) 集合 $\mathcal{C} \subset \Delta_{2^n}^0$ 称为一个不变子集, 如果

$$\mathbb{P}\left(\tilde{x}(t, w, x_0) \in \mathcal{C} \mid x_0 \in \mathcal{C}\right) = 1, \quad \forall t \geqslant 1.$$

(ii) 集合 $\mathcal{M} \subset \Delta_{2^n}^0$ 中所有不变子集的并, 称为 $\mathcal{M}$ 的最大不变子集, 记作 $\mathcal{I}(\mathcal{M})$.

下面这个结论是显见的.

引理 11.5.6 设 $\mathcal{C}$ 为网络 (11.5.13) 的不变子集, 那么, 从任何一点到 $\mathcal{C}$ 的概率是随时间单调增加的, 即

$$\mathbb{P}\left(\tilde{x}(k+1) \in \mathcal{C} \mid \tilde{x}_0 = \delta_{2^n}^j\right) \geqslant P_r\left\{\tilde{x}(k) \in \mathcal{C} \mid \tilde{x}_0 = \delta_{2^n}^j\right\}, \quad \forall j \in \Delta_{2^n}. \quad (11.5.29)$$

下面这个命题用来计算最大不变子集. 它可直接验证, 亦可由文献 [57] 的命题一推出.

命题 11.5.3 设 $\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}$ 为网络 (11.5.13) 的正常 (即在 Δ_{2^n} 上的) 概率转移矩阵. 设指标集 $\Lambda_0 \subset \{1, 2, \cdots, 2^n\}$ 且

$$\mathcal{M}_0 = \left\{\delta_{2^n}^j \mid j \in \Lambda_0\right\}.$$

定义一族指标集如下:

$$\Lambda_s = \left\{ j \in \Lambda_{s-1} \;\middle|\; \sum_{i \in \Lambda_{s-1}} \left[\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right]_{i,j} = 1 \right\}, \quad s = 1, 2, \cdots.$$

则必然存在一个 $\mathbf{k} \leqslant |\mathcal{M}_0|$, 使得

$$\Lambda_{\mathbf{k}+1} = \Lambda_{\mathbf{k}}.$$

并且,

$$\mathcal{I}\left(\mathcal{M}_0\right) = \left\{\delta_{2^n}^j \mid j \in \Lambda_{\mathbf{k}}\right\}.$$

为了证明下一个结论, 我们需要以下这个引理.

引理 11.5.7 [78]　对于一个实矩阵 $A \in \mathcal{M}_{n\times n}$,

$$\rho(A) \leqslant \|A\|_1, \tag{11.5.30}$$

这里, $\rho(A)$ 是 A 的谱半径, $\|A\|_1$ 是 A 的 ℓ_1 模, 即

$$\rho(A) = \max\{|\lambda| \mid \lambda \in \sigma(A)\}, \quad \|A\|_1 = \max_{1\leqslant j\leqslant n}\left\{\sum_{i=1}^{n}|A_{i,j}|\right\}.$$

引理 11.5.8　记 $\mathcal{A} = \mathcal{I}\left(\mathcal{M}_d^c\right) \cup \{\delta_{2^n}^0\}$, 则

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t,w,x_0) \in \mathcal{A}\right) = 1, \quad \forall x_0 \in \varDelta_{2^n}. \tag{11.5.31}$$

证明　显见, $\mathcal{A}$ 是一个不变集, 且 $\mathcal{A}^c = \varDelta_{2^n}^0 \backslash \mathcal{A}$ 不含任何非空不变子集. 不失一般性, 可设

$$\mathcal{I}\left(\mathcal{M}_d^c\right) = \left\{\delta_{2^n}^j \mid s+1 \leqslant j \leqslant 2^n\right\}.$$

将 $\mathcal{A}$ 当作一个点, 记为 δ_{s+1}^{s+1}, 将 $\delta_{2^n}^j$ 重新命名为 δ_{s+1}^j, $j = 1,2,\cdots,s$. 这样我们就可以构造一个新的随机逻辑动态系统, 记作 $\tilde{\Sigma}$, 这里, $\mathcal{A} = \delta_{s+1}^{s+1}$ 是其唯一的不动点. 设 $\tilde{\Sigma}$ 的 k 步转移概率矩阵为

$$\tilde{\mathbf{P}}(k) = \begin{bmatrix} \mathbf{\Gamma}(k) & \mathbf{0}_s \\ \alpha^{\mathrm{T}}(k) & 1 \end{bmatrix},$$

这里 $\mathbf{\Gamma}(k) \in \mathcal{M}_{s\times s}$, $\alpha(k) \in \mathbb{R}^s$. 因为 $\mathcal{I}\left(\mathcal{A}^c\right) = \varnothing$, 对 $1 \leqslant j \leqslant s$, 都存在一个正数 k_0^j, 使得 $\alpha_j(k_0^j) > 0$. 令 $k_0 = \max\limits_{1\leqslant j\leqslant s}\{k_0^j\}$, 由引理 11.5.6 可知 $\alpha_{k_0} > 0$. 因此有

$$\rho\left(\mathbf{\Gamma}(k_0)\right) \leqslant \|\mathbf{\Gamma}(k_0)\|_1 < 1.$$

因此, $\mathbf{\Gamma}(k_0)$ 是严 Schur 稳定的.

显然, (11.5.31) 成立, 当且仅当, 对 $\tilde{\Sigma}$ 任何状态转移到 δ_{s+1}^{s+1} 的概率趋于 1. 利用引理 11.5.3 和引理 11.5.6 可知, 这等价于证明

$$\lim_{r\to\infty} \alpha(rk_0) = \mathbf{1}_s. \tag{11.5.32}$$

记 $\mathbf{e}(rk_0) = \alpha(rk_0) - \mathbf{1}_s$, 利用公式

$$\tilde{\mathbf{P}}((r+1)k_0) = \tilde{\mathbf{P}}(rk_0)\tilde{\mathbf{P}}(k_0)$$

可得

$$\mathbf{e}((r+1)k_0) = \mathbf{\Gamma}^{\mathrm{T}}(k_0)\mathbf{e}(rk_0) + \left[\mathbf{\Gamma}^{\mathrm{T}}(k_0)\mathbf{1}_s + \alpha(k_0) - \mathbf{1}_s\right].$$

因为 $\tilde{\mathbf{P}}(k_0)$ 的每一列和为 1, 则得

$$\mathbf{e}((r+1)k_0) = \mathbf{\Gamma}^{\mathrm{T}}(k_0)\mathbf{e}(rk_0).$$

又因为 $\mathbf{\Gamma}(k_0)$ 是严 Schur 稳定的, 则得

$$\lim_{r\to\infty} \mathbf{e}(rk_0) = \mathbf{0}_s.$$

故 (11.5.32) 成立. □

命题 11.5.4 考察网络 (11.5.5), 设 $\Lambda_0 \subset \{1, 2, \cdots, 2^n\}$ 给定, 且

$$\mathcal{M}_0 = \{\delta_{2^n}^j \mid j \in \Lambda_0\}.$$

那么, $\mathcal{M}_d$ 是从 $\mathcal{M}_0$ 出发依分布渐近可达的, 当且仅当,

$$\left[\left(\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right)^{|\mathcal{M}_d^c|}\right]_{i,j} = 0, \quad \forall i \in \Lambda_{\mathcal{I}(\mathcal{M}_d^c)}, \quad \forall j \in \Lambda_0, \tag{11.5.33}$$

这里

$$\mathcal{I}\left(\mathcal{M}_d^c\right) = \left\{\delta_{2^n}^i \mid i \in \Lambda_{\mathcal{I}\left(\mathcal{M}_d^c\right)}\right\}, \text{ 且 } \Lambda_{\mathcal{I}\left(\mathcal{M}_d^c\right)} \subset \{1, 2, \cdots, 2^n\}.$$

证明 (必要性) 根据命题 11.5.1 (ii), $\mathcal{M}_d$ 从 $\mathcal{M}_0$ 依分布渐近可达, 当且仅当,

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t, w, x_0) = \delta_{2^n}^0\right) = 1, \quad \forall x_0 \in \mathcal{M}_0.$$

也就是说, 对任何 $x_0 = \delta_{2^n}^j$ 均成立:

$$\begin{aligned}
1 &= \lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t, w, \delta_{2^n}^j) = \delta_{2^n}^0\right) \\
&= \lim_{t\to\infty} \left[1 - \sum_{i=1}^{2^n} \mathbb{P}(\tilde{x}(t, w, \delta_{2^n}^j) = \delta_{2^n}^i)\right] \\
&= 1 - \lim_{t\to\infty} \sum_{i=1}^{2^n} \left[\left(\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right)^t\right]_{i,j}.
\end{aligned}$$

这等同于

$$\lim_{t\to\infty} \sum_{i=1}^{2^n} \left[\left(\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right)^t\right]_{i,j} = 0, \quad \forall j \in \Lambda_0$$

或

$$\lim_{t\to\infty} \mathrm{Col}_j \left[\left(\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right)^t\right] = \delta_{2^n}^0, \quad \forall j \in \Lambda_0. \tag{11.5.34}$$

因此, 我们只要证明 (11.5.34) 蕴涵 (11.5.33) 即可.

由 (11.5.34) 可知, 对任何初始状态 $\delta_{2^n}^j \in \mathcal{M}_0$ 有

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j)=\delta_{2^n}^0\right) = \lim_{t\to\infty}\left(1-\sum_{i=1}^{2^n}\left[\left(\mathbf{P}^X\mathbf{D}_{\mathcal{M}_d^c}\right)^t\right]_{i,j}\right)=1. \qquad (11.5.35)$$

于是有

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j)\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)=0,\quad \forall\delta_{2^n}^j\in\mathcal{M}_0. \qquad (11.5.36)$$

再由引理 11.5.6 可知

$$\mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j)\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)=0,\quad \forall\delta_{2^n}^j\in\mathcal{M}_0,\quad \forall t\geqslant 1.$$

再由引理 11.5.5 即得 (11.5.33).

(充分性) 由 (11.5.33) 可知

$$\mathbb{P}\left(\tilde{x}(|\mathcal{M}_d^c|,w,d_{2^n}^j)=\delta_{2^n}^i\right)=0,\quad \forall i\in\Lambda_{\mathcal{I}(\mathcal{M}_d^c)},\quad \forall j\in\Lambda_0.$$

于是有

$$\mathbb{P}\left(\tilde{x}(|\mathcal{M}_d^c|,w,\delta_{2^n}^j)\in\mathcal{I}(\mathcal{M}_d^c)\right)=0,\quad \forall j\in\Gamma_0.$$

由引理 11.5.6 可知, 对任意初态 $\delta_{2^n}^j\in\mathcal{M}_0$ 有

$$\mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j)\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)=0,\quad \forall 1\leqslant t\leqslant|\mathcal{M}_d^c|. \qquad (11.5.37)$$

其次我们证明

$$\mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j)\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)=0,\quad \forall j\in\Lambda_0,\quad \forall t>|\mathcal{M}_d^c|. \qquad (11.5.38)$$

设有 $T_0>|\mathcal{M}_d^c|$ 及 $j_0\in\Lambda_0$, 使得

$$\mathbb{P}\left(\tilde{x}(T_0,w,\delta_{2^n}^{j_0})\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)>0,$$

即存在一条长度为 T_0 的容许轨道从 $\delta_{2^n}^{j_0}$ 到达 $\mathcal{I}\left(\mathcal{M}_d^c\right)$. 则由引理 11.5.5 可知, 必有一条长度为 $T_1\leqslant|\mathcal{M}_d^c|$ 的容许轨道从 $\delta_{2^n}^{j_0}$ 到达 $\mathcal{I}\left(\mathcal{M}_d^c\right)$. 因此,

$$\mathbb{P}\left(\exists T_1\leqslant|\mathcal{M}_d^c|,\text{使得 }\tilde{x}(T_1,w,\delta_{2^n}^{j_0})\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)>0,$$

这与 (11.5.37) 矛盾.

由 (11.5.37) 及 (11.5.38) 可得

$$\mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j)\in\mathcal{I}\left(\mathcal{M}_d^c\right)\right)=0,\quad \forall j\in\Lambda_0,\quad \forall t\geqslant 1. \qquad (11.5.39)$$

于是有

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j) \in \mathcal{I}(\mathcal{M}_d^c)\right) = 0, \quad \forall j \in \Lambda_0.$$

根据引理 11.5.8 即得

$$\lim_{t\to\infty} \mathbb{P}\left(\tilde{x}(t,w,\delta_{2^n}^j) = \delta_{2^n}^0\right) = 1, \quad \forall j \in \Lambda_0. \tag{11.5.40}$$

即, $\delta_{2^n}^0$ 是从 $\mathcal{M}_0$ 出发, 依分布渐近可达. □

例 11.5.2　回顾例 11.5.1, 不难算得

$$\left(\mathbf{P}^X \mathbf{D}_{\mathcal{M}_d^c}\right)^6 = 10^{-5} \times \begin{bmatrix} 0 & 0 & 0 & 6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 30 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 17 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 34 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

(i)　根据命题 11.5.2 (i), 该网络是从任何 $\mathcal{M}_0 \subset \varDelta_{2^n} \backslash \{\delta_8^4\}$ 在时间 $0 \leqslant t \leqslant 5$ 依概率 1 到达 $\mathcal{M}_d$.

(ii)　根据命题 11.5.3, $\mathcal{I}\left(\mathcal{M}_d^c\right) = \varnothing$.

(iii)　根据命题 11.5.4, 对于该网络, $\mathcal{M}_d$ 是从任何子集出发, 依分布渐近可达.

11.5.5　能观性的检测

考察概率布尔网络 (11.5.5), 类似于确定型布尔网络能观性的辅助系统方法, 构造一个影子系统

$$\begin{cases} \bar{x}(t+1) = \mathbf{L} w(t) \bar{x}(t), \\ \bar{y}(t) = H \bar{x}(t). \end{cases} \tag{11.5.41}$$

将 (11.5.5) 与 (11.5.41) 放到一起, 则得

$$\begin{cases} \xi(t+1) = \mathbf{L}_\xi w(t) \xi(t), \\ \eta(t) = H_\eta \xi(t), \end{cases} \tag{11.5.42}$$

这里, $\xi(t) = x(t) \ltimes \bar{x}(t)$, $\eta(t) = y(t) \ltimes \bar{y}(t)$, 以及

$$\begin{cases} \mathbf{L}_\xi = (\mathbf{L} * \mathbf{L}) \left(I_N \otimes W_{[N,2^n]}\right) \mathrm{PR}_N, \\ H_\eta = H * H, \end{cases}$$

这里, “$*$” 是 Khatri-Lao 积, PR_N 是降阶矩阵.

令 $\xi_0 = x_0 \ltimes \bar{x}_0$, 则从构造可知

$$\xi(t,w,\xi_0) = x(t,w,x_0) \ltimes \bar{x}(t,w,\bar{x}_0), \quad t \geqslant 0.$$

记

$$\begin{aligned} \mathcal{G}_I &:= \{x \ltimes \bar{x} \mid x \neq \bar{x}\}, \\ \mathcal{G}_H &:= \{x \ltimes \bar{x} \mid Hx \neq H\bar{x}\}, \\ \mathcal{G}_{I\backslash H} &:= \{x \ltimes \bar{x} \in \mathcal{G}_I \backslash \mathcal{G}_H \mid \Lambda_x < \Lambda_{\bar{x}}\}. \end{aligned}$$

注意到 x 与 $\bar{x}$ 的对称性, 在考虑能观性时无须区分 $x \ltimes \bar{x}$ 与 $\bar{x} \ltimes x$, 因此, 集合 $\mathcal{G}_{I\backslash H}$ 中只取 $\mathcal{G}_I \backslash \mathcal{G}_H$ 的一半.

命题 11.5.5　考察网络 (11.5.5) 与网络 (11.5.42).

(i)　(11.5.5) 在时间 $0 \leqslant t \leqslant T$ 内依概率 1 可观, 当且仅当, (11.5.42) 的子集 $\mathcal{G}_H$ 在时间 $0 \leqslant t \leqslant T$ 内依概率 1 从 $\mathcal{G}_{I\backslash H}$ 可达;

(ii)　(11.5.5) 在有限时间内依概率 1 可观, 当且仅当, (11.5.42) 的子集 $\mathcal{G}_H$ 在有限时间依概率 1 从 $\mathcal{G}_{I\backslash H}$ 可达;

(iii)　(11.5.5) 依分布渐近可观, 当且仅当, (11.5.42) 的子集 $\mathcal{G}_H$ 从 $\mathcal{G}_{I\backslash H}$ 依分布渐近可达.

证明　对任给的 x_0, $\bar{x}_0 \in \varDelta_{2^n}$, 记相应输出轨线为 $\mathbf{y}(t,w,x_0)$ 及 $\mathbf{y}(t,w,\bar{x}_0)$, 我们有

$$\begin{aligned} &\mathbb{P}\left(y(t,w,x_0) \neq y(t,w,\bar{x}_0)\right) \\ =\ &\mathbb{P}\left(Hx(t,w,x_0) \neq H\bar{x}(t,w,\bar{x}_0)\right) \\ =\ &\mathbb{P}\left(\bigcup_{j=0}^{t}[Hx(j,w,x_0) \neq H\bar{x}(j,w,\bar{x}_0)]\right) \\ =\ &\mathbb{P}\left(\bigcup_{j=0}^{t}[x(j,w,x_0) \ltimes \bar{x}(j,w,\bar{x}_0) \in \mathcal{G}_H]\right) \\ =\ &\mathbb{P}\left([\xi_0 \in \mathcal{G}_H] \cup [\xi(1,w,\xi_0) \in \mathcal{G}_H] \cup \cdots \cup [\xi(t,w,\xi_0) \in \mathcal{G}_H]\right) \\ =\ &\mathbb{P}\left(\exists 0 \leqslant k \leqslant t, \text{使得 } \xi(k,w,\xi_0) \in \mathcal{G}_H\right). \end{aligned} \tag{11.5.43}$$

由式 (11.5.43) 不难看出, 对任何 $\xi_0 \in \mathcal{G}_{I\backslash H}$,

(1)　(11.5.7) 等价于

$$\mathbb{P}\left(\exists 0 \leqslant k \leqslant T, \text{使得 } \xi(k,w,\xi_0) \in \mathcal{G}_H\right) = 1. \tag{11.5.44}$$

(2) (11.5.8) 等价于

$$\lim_{t\to\infty} \mathbb{P}\left(\exists 0 \leqslant k \leqslant t, \text{使得 } \xi(k,w,\xi_0) \in \mathcal{G}_H\right) = 1. \tag{11.5.45}$$

于是可知 (i) 与 (iii) 成立. 利用 (i) 与 (iii) 即可推出 (ii). □

利用命题 11.5.5 不难得几个能观性的主要结果:

定理 11.5.1 考察网络 (11.5.5). 设 $\mathcal{G}_H = \{\delta_{2^{2n}}^i \mid i \in \Lambda_H\}, \mathcal{G}_{I\backslash H} = \left\{\delta_{2^{2n}}^j \mid j \in \Lambda_{I\backslash H}\right\}$.

(i) (11.5.5) 在 $0 \leqslant t \leqslant T$ 时间内依概率 1 可观, 当且仅当,

$$\mathrm{Col}_j\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{T+1}\right] = \delta_{2^n}^0, \quad \forall j \in \Lambda_{I\backslash H}. \tag{11.5.46}$$

(ii) (11.5.5) 在有限时间内以概率 1 可观, 当且仅当,

$$\mathrm{Col}_j\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{|\mathcal{G}_H^c|+1}\right] = \delta_{2^n}^0, \quad \forall j \in \Lambda_{I\backslash H}. \tag{11.5.47}$$

证明 (i) 根据命题 11.5.5 (i), (11.5.5) 在时间 $0 \leqslant t \leqslant T$ 内依概率 1 可观, 当且仅当, (11.5.42) 的集合 $\mathcal{G}_H$ 是从集合 $\mathcal{G}_{I\backslash H}$ 出发, 在时间 $0 \leqslant t \leqslant T$ 内依概率 1 可达. 因此, (i) 可由命题 11.5.2 (i) 直接推出.

(ii) 根据命题 11.5.5 (iii), (11.5.5) 在有限时间内依概率 1 可观, 当且仅当, (11.5.42) 的集合 $\mathcal{G}_H$ 是从集合 $\mathcal{G}_{I\backslash H}$ 出发, 在有限时间内依概率 1 可达. 因此, (ii) 可由命题 11.5.2 (ii) 直接推出. □

因为有限时间依概率可观是有限时间依概率 1 可观的一种推广, 类似的讨论即可得到以下结果.

推论 11.5.1 考察网络 (11.5.5).

(i) (11.5.5) 在 $0 \leqslant t \leqslant T$ 时间内依概率可观, 当且仅当,

$$\sum_{i=1}^{2^{2n}}\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{T+1}\right]_{i,j} < 1, \quad \forall j \in \Lambda_{I\backslash H}. \tag{11.5.48}$$

(ii) (11.5.5) 在有限时间内依概率可观, 当且仅当,

$$\sum_{i=1}^{2^{2n}}\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{|\mathcal{G}_H^c|+1}\right]_{i,j} < 1, \quad \forall j \in \Lambda_{I\backslash H}. \tag{11.5.49}$$

定理11.5.2　考察网络 (11.5.5). 设 $\mathcal{G}_H=\{\delta_{2^{2n}}^i \mid i\in\Lambda_H\}, \mathcal{G}_{I\backslash H}=\left\{\delta_{2^{2n}}^j \mid j\in\Lambda_{I\backslash H}\right\}$. 定义一族指标集如下:

$$\Lambda_0=\Lambda_H^c,$$

$$\Lambda_s=\left\{j\in\Lambda_{s-1}\left|\sum_{i\in\Lambda_{s-1}}\left[\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right]_{i,j}=1\right.\right\},\quad s=1,2,\cdots,$$

这里 $\mathcal{G}_H^c=\left\{\delta_{2^{2n}}^k \mid k\in\Lambda_H^c\right\}$. 设 $\mathbf{k}$ 为最小的 k, 使得 $\Lambda_k=\Lambda_{k+1}$. 那么, (11.5.5) 是依分布渐近可观, 当且仅当,

$$\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{|\mathcal{G}_H^c|}\right]_{i,j}=0,\quad \forall i\in\Lambda_{\mathbf{k}},\quad \forall j\in\Lambda_{I\backslash H}. \tag{11.5.50}$$

证明　根据命题 11.5.5 (iii), (11.5.5) 依分布渐近可观, 当且仅当, (11.5.42) 的集合 $\mathcal{G}_H$ 是从 $\mathcal{G}_{I\backslash H}$ 分布渐近可达. 再由命题 11.5.4 即得结论. □

本章讨论了三种能观性:

I: 有限时间依概率 1 可观;

II: 依分布渐近可观;

III: 有限时间依概率可观.

不难看出

$$\text{I}\ \Rightarrow\ \text{II}\ \Rightarrow\ \text{III}.$$

但是, 相反方向是不成立的, 即其中没有两个是等价的. 本节给出三个例子: 例 11.5.3 是有限时间依概率 1 可观; 例 11.5.4 是依分布渐近可观, 但不是有限时间依概率 1 可观; 例 11.5.5 是有限时间依概率可观. 但不是依分布渐近可观.

图 11.5.1 展示了它们的关系.

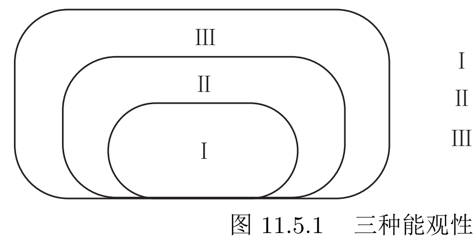


图 11.5.1　三种能观性

例 11.5.3　考察概率布尔网络

$$\begin{cases}x(t+1)=L_{\sigma(t)}x(t),\\ y(t)=Hx(t),\end{cases} \tag{11.5.51}$$

这里, $x(t) \in \Delta_{2^n}$, $n=3$, $y(t) \in \Delta_{2^q}$, $q=2$, $\sigma(t) \in \Lambda = \{1,2,3,4,5,6\}$,

$$\begin{aligned} L_1 &= \delta_8[1,2,1,1,5,8,6,8], \\ L_2 &= \delta_8[6,4,5,8,6,8,3,8], \\ L_3 &= \delta_8[1,4,5,1,6,8,6,8], \\ L_4 &= \delta_8[6,2,1,8,5,8,3,8], \\ L_5 &= \delta_8[1,2,5,1,5,8,3,8], \\ L_6 &= \delta_8[1,4,1,8,5,8,6,8]. \end{aligned}$$

$$\mathbf{L} = [L_1, L_2, L_3, L_4, L_5, L_6].$$

$$H = \delta_4[1,2,2,3,3,1,2,4].$$

$$p_1 = 0.3, \quad p_2 = 0.2, \quad p_3 = 0.15, \quad p_4 = 0.1, \quad p_5 = 0.1, \quad p_6 = 0.15.$$

不难算出

$$\mathcal{G}_H = \left\{ \delta_{64}^j \mid j \in \Lambda_H \right\}, \quad \mathcal{G}_{I\backslash H} = \left\{ \delta_{64}^j \mid j \in \Lambda_{I\backslash H} \right\},$$

这里,

$$\begin{aligned} \Lambda_H = & \{2,3,4,5,7,8,9,12,13,14,16,17,20,21,22,24,25, \\ & 26,27,30,31,32,33,34,35,38,39,40,42,43,44,45, \\ & 47,48,49,52,53,54,56,57,58,59,60,61,62,63\}; \\ \Lambda_{I\backslash H} = & \{6,11,15,23,29\}. \end{aligned}$$

$\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}$ 见文献 [151]. 于是可计算

$$\mathrm{Col}_j\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^5\right] = \delta_{64}^0, \quad \forall j \in \Lambda_{I\backslash H}.$$

根据定理 11.5.1 可知, (11.5.51) 是有限时间依概率 1 可观, 因此也是依分布渐近可观, 以及有限时间依概率可观的.

例 11.5.4 考察网络

$$\begin{cases} x(t+1) = L_{\sigma(t)} x(t), \\ y(t) = Hx(t), \end{cases} \tag{11.5.52}$$

其中, $n=3$, $q=2$, $\Lambda = \{1,2,3,4\}$.

$$L_1 = \delta_8[1,3,2,4,5,7,6,5],$$

$$L_2 = \delta_8[3,4,5,6,7,5,2,8],$$

$$L_3 = \delta_8[1,8,3,7,5,4,1,3],$$

$$L_4 = \delta_8[4,6,8,1,5,7,5,6].$$

$$p_1 = 0.4, \quad p_2 = 0.1, \quad p_3 = 0.2, \quad p_4 = 0.3.$$

$$H = \delta_4[1,3,1,4,2,3,4,4].$$

$$\begin{aligned}\Lambda_H = \ & \{2,4,5,6,7,8,9,11,12,13,15,16,18,20,21,22,23,24,\\ & 25,26,27,29,30,33,34,35,36,38,39,40,41,43,\\ & 44,45,47,48,49,50,51,53,54,57,58,59,61,62\};\\ \Lambda_{I\backslash H} = \ & \{3,14,31,32,56\}.\end{aligned}$$

因此,

$$|\mathcal{G}_H^c| = 64 - |\Lambda_H| = 18.$$

直接计算可得

$$\mathrm{Col}_i\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{19}\right] \neq \delta_{64}^0, \quad i = 3, 56.$$

因此, 根据定理 11.5.1, (11.5.52) 不是有限时间依概率 1 可观.

利用命题 11.5.3 可得

$$\mathcal{I}\left(\mathcal{G}_H^c\right) = \left\{\delta_{64}^k \mid k \in \Lambda_{k^*}\right\} = \delta_{64}\{1,10,19,28,37,46,55,64\}.$$

并且, 不难计算

$$\left[\left(\mathbf{P}^{\xi}\mathbf{D}_{\mathcal{G}_H^c}\right)^{18}\right]_{i,j} = 0, \quad \forall j \in \Lambda_{I\backslash H}.$$

因此, 根据定理 11.5.2, (11.5.52) 是依分布渐近可观的, 因此也是有限时间依概率可观的.

例 11.5.5 考察网络

$$\begin{cases} x(t+1) = L_{\sigma(t)}x(t), \\ y(t) = Hx(t), \end{cases} \tag{11.5.53}$$

其中, $n = 3$, $q = 2$, $\Lambda = \{1,2,3,4\}$.

$$L_1 = \delta_8[1,2,2,4,5,6,6,2],$$

$$L_2 = \delta_8[1,1,2,2,6,7,6,8],$$

$$L_3 = \delta_8[1,2,1,4,1,6,6,8],$$

$$L_4 = \delta_8[1,1,3,3,5,5,6,8].$$

$$p_1 = 0.2, \quad p_2 = 0.15, \quad p_3 = 0.2, \quad p_4 = 0.1.$$

$$H = \delta_4[1, 2, 1, 3, 1, 4, 3, 3].$$

$$\begin{aligned} \Lambda_H = \ & \{2, 4, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16, 18, 20, 22, 23, \\ & 24, 25, 26, 27, 29, 30, 34, 36, 38, 39, 40, 41, 42, 43, \\ & 44, 45, 47, 48, 49, 50, 51, 53, 54, 57, 58, 59, 60, 61\}; \\ \Lambda_{I\backslash H} = \ & \{3, 5, 21, 31, 32, 56\}. \end{aligned}$$

因此,

$$|\mathcal{G}_H^c| = 64 - |\Lambda_H| = 20.$$

直接计算可得

$$\mathrm{Col}_j \left[\left(\mathbf{P}^{\xi} \mathbf{D}_{\mathcal{G}_H^c} \right)^{21} \right] \neq \delta_{64}^0, \quad j = 3, 5, 21, 32.$$

因此, 根据定理 11.5.1, (11.5.53) 不是有限时间依概率 1 可观.

利用命题 11.5.3 可得

$$\mathcal{I}\left(\mathcal{G}_H^c\right) = \left\{ \delta_{64}^k \mid k \in \Lambda_{k^*} \right\} = \delta_{64}\{1, 10, 19, 28, 37, 46, 55, 64\}.$$

并且, 不难计算

$$\left[\mathbf{P}^{\xi} \mathbf{D}_{\mathcal{G}_H^c} \right]_{1,3} = 0.2.$$

因此, 根据定理 11.5.2, (11.5.53) 不是依分布渐近可观的.

直接计算可得

$$\sum_{i=1}^{64} \left[\left(\mathbf{P}^{\xi} \mathbf{D}_{\mathcal{G}_H^c} \right)^{20} \right]_{i,j} < 1, \quad j = 3, 5, 21, 31, 32, 56.$$

因此, 根据推论 11.5.1, (11.5.53) 是有限时间依概率可观的.

第 12 章　最 优 控 制

最优控制是控制系统研究中最基本的问题之一, 对于逻辑控制系统也一样. 从布尔网络控制的半张量积方法出现开始, 布尔网络及一般逻辑系统的优化控制就成为一个研究的核心问题. 考虑的问题包括: 布尔网络的优化控制[45, 137], 时间最优控制[81], 混合值逻辑系统的最优控制[33], 随机布尔网络的优化控制[121], 最优控制方法的工业应用[122], 等等.

12.1　时间最优控制

时间最优控制是逻辑系统优化控制中最简单的一种, 它无须目标函数, 因为时间就是天生的目标函数.

考察一个 k 值逻辑控制网络

$$\begin{cases} X_1(t+1) = F_1(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)) \\ X_2(t+1) = F_2(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)) \\ \qquad\qquad\cdots\cdots \\ X_n(t+1) = F_n(X_1(t),\cdots,X_n(t),U_1(t),\cdots,U_m(t)), \end{cases} \tag{12.1.1}$$

这里, $X_i, U_j \in \mathcal{D}_k$, X_i 为状态变量, U_j 为控制变量, F_i 为 k 值逻辑函数. (12.1.1) 简记为

$$X(t+1) = F(X(t), U(t)), \tag{12.1.2}$$

这里, $X = (X_1,\cdots,X_n)$, $U = (U_1,\cdots,U_m)$. 当 $k=2$ 时, (12.1.1) 成为一个布尔控制网络. 本章直接讨论一般的 k.

定义 12.1.1　*考察 k 值逻辑控制网络 (12.1.1).*

给定初始值 X_0 和终点值 X_d, 时间最优控制是指找到控制序列 $U(t), t = 0,1,2,\cdots,T^*-1$, 使得 $X(0)=X_0$, $X(T^*)=X_d$, 并且

$$T^* = \min\{T \mid \text{存在 } U(0),\cdots,U(T-1), \text{使得 } X(0)=X_0, X(T)=X_d\}. \tag{12.1.3}$$

首先, 本节计划用第 8 章提出的 Ledley 前提解的方法寻找状态反馈时间最优控制. 正如第 8 章指出的, 任何一个 (自由序列) 控制, 都有等效的状态反馈控制. 因此, 不失一般性, 只要寻找状态反馈时间最优控制就够了. 其次, 要考虑 X_0 到 X_d 的时间最优控制问题, 前提是系统从 X_0 到 X_d 可控. 但是下面的算法可同时检验可控性和提供最优控制.

算法 12.1.1 第 0 步 (初始步): 设初始的容许集合为 $W_0 = \Omega(0) = \{X_d\}$, 这里, $X_d = (C_1, C_2, \cdots, C_n)$, 构造真值矩阵 $T_{\Omega(0)}$, 利用它找出对应于 $\Omega(0)$ 的极大集, 记作 $W_1 = \mathrm{Max}(\Omega(0))$. 由 $T_{\Omega(0)}$ 构造 $U(0)$. 检验是否 $X^d \in W_1$?

(i) 如果 $X_0 \in W_1$, 则 X_d 可从 X_0 出发, 利用控制 $U(0)$ 一步到达. 算法结束.

(ii) 令 $k = 1$, 进入循环步骤.

第 k (循环步骤, $k \geqslant 1$) : 令

$$\Omega(k) = W_k \backslash \{W_{k-1}, \cdots, W_0\}.$$

(i) 如果 $\Omega(k) = \varnothing$, 则时间最优控制问题无解, 算法结束.

(ii) 否则 (即 $\Omega(k) \neq \varnothing$), 构造真值矩阵 $T_{\Omega(k)}$, 依此构造 $U(k)$. 利用 $T_{\Omega(k)}$ 找出关于 $\Omega(k)$ 的极大集 $W_{k+1} = \mathrm{Max}(\Omega(k))$. 然后检验: 如果 $X_0 \in W_{k+1}$, 则 $U(0), U(1), \cdots, U(k)$ 使 $X(k+1) = X_d$, 并且这一控制序列是最小时间控制. 如果 $X_0 \notin W_{k+1}$, 则令 $k = k+1$, 返回循环步骤.

定理 12.1.1 (i) 如果算法 12.1.1 可产生有效控制序列, 使轨线到达 X_d, 则该控制序列为最小时间控制.

(ii) 如果算法 12.1.1 不能产生有效控制序列, 使轨线到达 X_d, 则该系统不能由 X_0 控制至 X_d.

证明 由算法可知, 如果 $X_d \notin W_s$, $s < t$, 则不存在状态反馈控制 (亦即任何控制), 使轨线从 X_0 出发, 在 $t-1$ 步内到达 X_d, (ii) 显见. 现在如果 $X_d \notin W_t$, 则存在控制序列, 使轨线从 X_0 出发, 在第 t 步内到达 X_d. 结合 (ii) 可知, t 为最小到达时间. 即得 (i). □

下面考虑一个例子.

例 12.1.1 考察布尔控制网络

$$\begin{cases} X_1(t+1) = X_2(t) \vee U(t), \\ X_2(t+1) = X_3(t), \\ X_3(t+1) = X_1(t) \wedge X_4(t), \\ X_4(t+1) = \neg U(t). \end{cases} \tag{12.1.4}$$

(i) 寻找从 $X_0=(1,1,1,0)$ 到 $X_d=(0,1,1,1)$ 的时间最优控制.

先考虑 (12.1.3) 的代数状态空间表达式, 它是

$$x(t+1)=Lu(t)x(t), \tag{12.1.5}$$

这里, $x(t)=\ltimes_{i=1}^4 x_i(t)$,

$$\begin{aligned}L=\delta_{16}[&2,4,6,8,2,4,6,8,4,4,8,8,4,4,8,8,\\&1,3,5,7,9,11,13,15,3,3,7,7,11,11,15,15].\end{aligned} \tag{12.1.6}$$

容易算得 $x_0=\vec{X}_0=\delta_{16}^2$, $x_d=\vec{X}_d=\delta_{16}^9$. 利用算法 12.1.1, 计算如下:

$$\begin{aligned}W_0&=\Omega(0):=\{\delta_{16}^9\},\\W_1&=\mathrm{Max}(\Omega(0))=\{\delta_{16}^5\},\\\Omega(1)&=W_1\backslash W_0=\{\delta_{16}^5\},\\W_2&=\mathrm{Max}(\Omega(1))=\{\delta_{16}^3\},\\\Omega(2)&=W_2\backslash(W_1\cup W_0)=\{\delta_{16}^3\},\\W_3&=\mathrm{Max}(\Omega(2))=\{\delta_{16}^2,\delta_{16}^9,\delta_{16}^{10}\},\\\Omega(3)&=W_3\backslash(W_2\cup W_1\cup W_0)=\{\delta_{16}^2,\delta_{16}^{10}\}.\end{aligned} \tag{12.1.7}$$

因为

$$\delta_{16}^2\in W_3,$$

则最优 (最小) 控制时间为 $T^*=3$.

下面寻找最优反馈控制 $u^*(t)=Kx(t)$. 考察 $T_{\Omega(0)}$:

$$T_{\omega(0)}=\begin{bmatrix}0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\end{bmatrix},$$

可知

$$\mathrm{Col}_5(K)=\delta_2^2.$$

考察

$$T_{\omega(1)}=\begin{bmatrix}0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\end{bmatrix},$$

可知

$$\mathrm{Col}_3(K)=\delta_2^2.$$

考察

$$T_{\omega(2)}=\begin{bmatrix}0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&1&0&0&0&0&0&0&0\end{bmatrix},$$

可知

$$\mathrm{Col}_2(K)=\delta_2^2;\quad \mathrm{Col}_{10}(K)=\delta_2^2.$$

综上可知

$$K=\delta_2[*,2,2,*,2,*,*,*,*,2,*,*,*,*,*,*]. \tag{12.1.8}$$

这里, $*$ 可任取.

(ii) 寻找从 $X_0=(1,1,0,0)$ 到 $X_d=(0,1,1,1)$ 的时间最优控制.

因 $x_0=\vec{X}_0=\delta_{16}^4$ 未出现于 (12.1.6) 的 W_0,W_1 及 W_2 中, 我们需继续算法 12.1.1.

$$\begin{aligned} W_4&=\mathrm{Max}(\Omega(3))=\{\delta_{16}^1\},\\ \Omega(4)&=\{\delta_{16}^1\},\\ W_5&=\mathrm{Max}(\Omega(4))=\{\delta_{16}^1\},\\ \Omega(5)&=W_5\backslash(W_4\cup W_3\cup W_2\cup W_1\cup W_0)=\varnothing. \end{aligned} \tag{12.1.9}$$

故从 $X_0=(1,1,0,0)$ 到 $X_d=(0,1,1,1)$ 的时间最优控制无解.

注 12.1.1 回顾例 12.1.1,

(i) 不难看出, 能到达 X_d (即 $x_d=\delta_{16}^9$) 的点包括 $\delta_{16}\{1,2,3,5,10\}$, 相应状态反馈控制为 $u^*(t)=Kx(t)$, 这里,

$$K=\delta_2[1,2,2,*,2,*,*,*,*,2,*,*,*,*,*,*]. \tag{12.1.10}$$

(ii) 布尔控制网络 12.1.4, 其能控性矩阵 $\mathcal{C}$ 为

$$\begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&0&1&0&0&0&1&1&0&0&0&0&0&0\\
1&1&1&0&1&0&0&0&1&1&0&0&0&0&0&0\\
1&1&1&1&1&1&1&0&1&1&1&1&1&1&0&0\\
1&1&1&0&1&0&0&0&1&1&0&0&0&0&0&0\\
1&1&1&1&1&1&1&0&1&1&1&1&1&1&0&0\\
1&1&1&1&1&1&1&0&1&1&1&1&1&1&0&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
1&1&1&0&1&0&0&0&1&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&0&1&1&1&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&0&1&1&1&1&1&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix},$$

从上式第 9 行可知, $\delta_{16}\{1,2,3,5,10\}$ 可达 δ_{16}^{9}. 由此可见, 算法 12.1.1 可自动验证能控性.

12.2　平均值最优控制

12.2.1　输入-状态转移图

考察逻辑控制网络 (12.1.1), 利用代数状态空间表示可得

$$x(t+1) = Lu(t)x(t), \tag{12.2.1}$$

这里, $x_i(t) = \vec{X}_i(t) \in \Delta_k$, $x(t) = \ltimes_{i=1}^{n} x_i(t) \in \Delta_{k^n}$, $u_j(t) = \vec{U}_j(t) \in \Delta_k$, $u(t) = \ltimes_{j=1}^{m} u_j(t) \in \Delta_{k^m}$, $L \in \mathcal{L}_{k^n \times k^{m+n}}$.

设系统的优化的目标函数, 或称支付函数, 在每个时刻 t 为

$$P(t) = P(X(t), U(t)), \quad t = 0, 1, 2, \cdots, \tag{12.2.2}$$

这里, $P : \mathcal{D}_k^n \times \mathcal{D}_k^m \to \mathbb{R}$.

利用 $X(t), U(t)$ 的向量表示, 目标函数可表示为 $P(x(t), u(t)) : \Delta_{k^n} \times \Delta_{k^m} \to \mathbb{R}$. 记从 x_0 出发, 在 $u(t)$ 作用下系统轨线为 $x(t, x_0, u)$ (简记为 $x(t)$). 则 $x(t, x_0, u)$ 的平均支付定义为

$$J(x(t, x_0, u)) = J(u) = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} P(x(t), u(t)). \tag{12.2.3}$$

最优控制是指找出控制 $u^*(t)$ 的极大化目标函数 $J(u)$, 即

$$J(u^*) = \max_{u} J(u). \tag{12.2.4}$$

在最优控制的研究中, 以下几个问题必须解决: ① 最优控制 $u^*(t)$ 是否存在? ② 如果最优控制存在, 它是否唯一? ③ 如何设计最优控制? 这是我们所要解决的问题.

记控制–状态乘积空间为 $\mathcal{S} = \mathcal{D}_k^n \times \mathcal{D}_k^m$ (在向量形式下有 $\mathcal{S} = \Delta_{k^{m+n}}$). 下面的定义来自文献 [103].

定义 12.2.1　$\mathcal{S}$ 上的一个有向图称为网络 (12.1.1) 的输入–状态转移图, 如果它依如下方式形成:

(i)　其结点为 $\mathcal{S} = \{(U, X) \mid X \in \mathcal{D}_{2^n}, U \in \mathcal{D}_{2^m}\}$;

(ii)　$S_p = (U_p, X_p) \in \mathcal{S}$, $S_q = (U_q, X_q) \in \mathcal{S}$, $\overrightarrow{S_p S_q}$ 为一条有向边, 当且仅当,

$$X_q = F(U_p, X_p).$$

亦可用向量形式 $s_p = (u_p, x_p) = (\delta_{k^m}^{\alpha}, \delta_{k^n}^{\beta})$ 表示结点. 它可进一步简记为 $s_p = u_p x_p = \delta_{k^{m+n}}^{\gamma} \in \Delta_{k^{m+n}}$, 因为 s_p 具有唯一分解 (u_p, x_p). 记 $\delta_{k^{m+n}}^{\gamma} = \delta_{k^m}^{\alpha}\delta_{k^n}^{\beta}$, 则

$$\gamma = (\alpha - 1)k^n + \beta,$$

这里,

$$\alpha = \left[\frac{\gamma}{k^n}\right] + 1, \quad \beta = \gamma \pmod{k^n}.$$

在代数表达式下, $\overrightarrow{S_p S_q}$ (或 (s_p, s_q)) 是一条有向边, 当且仅当,

$$x_q = L u_p x_p \quad (\text{或 } x_q = L s_p).$$

设 $s_p = \delta_{k^{m+n}}^{\xi}$ 及 $s_q = \delta_{k^{m+n}}^{\eta}$, 则边 (s_p, s_q) 亦可表达为

$$\delta_{k^{m+n}}(\xi, \eta).$$

输入–状态转移图的拓扑在优化控制中起着重要作用.

定义 12.2.2 (i) 一个结点 $s' \in \mathcal{S}$ 称为从 $s = (u, x) \in \mathcal{S}$ 可达的, 如果存在从 s 到 s' 的一条边. s 的可达点集记为 $R(s)$.

(ii) 输入–状态转移图称为强连通的, 如果每一个点 $s \in \mathcal{S}$ 均满足

$$R(s) = \mathcal{S}.$$

我们用一个例子描述输入–状态转移图.

例 12.2.1 考察布尔控制网络

$$\begin{cases} X_1(t+1) = (X_1(t) \wedge U(t)) \vee X_2(t), \\ X_2(t+1) = (U(t) \wedge X_1(t) \wedge \neg X_2(t)) \vee (\neg U(t) \wedge X_1(t)). \end{cases} \tag{12.2.5}$$

其代数状态空间表达式为

$$x(t+1) = L u(t) x(t), \tag{12.2.6}$$

这里,

$$L = \delta_4[2, 1, 2, 4, 1, 3, 2, 4].$$

该网络的输入–状态转移图有八个结点:

$$\begin{aligned} &A = (\delta_2^1, \delta_4^1), \quad B = (\delta_2^1, \delta_4^2), \quad C = (\delta_2^1, \delta_4^3), \quad D = (\delta_2^1, \delta_4^4), \\ &E = (\delta_2^2, \delta_4^1), \quad F = (\delta_2^2, \delta_4^2), \quad G = (\delta_2^1, \delta_4^3), \quad H = (\delta_2^1, \delta_4^4). \end{aligned}$$

(12.2.5) 的输入–状态转移图见图 12.2.1.

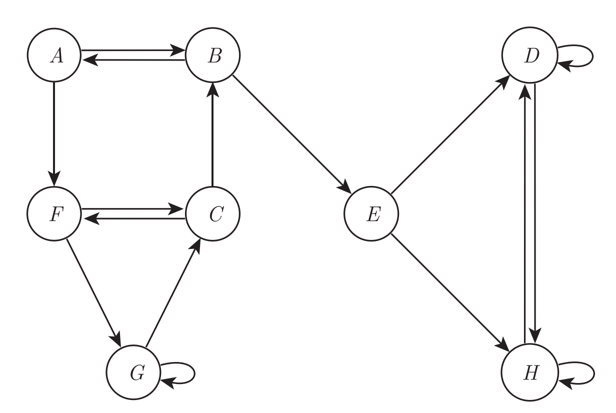


图 12.2.1 (12.2.5) 的输入–状态转移图

从图中可见

$$R(A) = R(B) = R(C) = R(F) = R(G) = \mathcal{S}.$$

$$R(E) = \{D, H\}, \quad R(H) = \{H, D\}, \quad R(D) = \{D, H\},$$

显然. 它不是强连通的.

实际上, 强连通等价于完全能控.

定义 12.2.3 (i) 设 $s_i \in \mathcal{S}$, $i = 1, 2, \cdots, \ell$. $(s_1, s_2, \cdots, s_\ell)$ 称为一条路径, 如果 (s_i, s_{i+1}), $i = 1, \cdots, \ell - 1$ 为输入–状态转移图的边.

(ii) 一条路径 $(s_1, s_2, \cdots, s_{\ell+1})$ 称为一个环, 如果 $s_i \neq s_j$, $1 \leqslant i \neq j \leqslant \ell$, 并且, $s_{\ell+1} = s_1$, ℓ 称为环的长度. 特别是, 如果 $\ell = 1$, 则环称为不动点.

(iii) 设 $C = (s_1, s_2, \cdots, s_{\ell+1})$ 为一环, 记 $s_i = (u_i, x_i), i = 1, \cdots, \ell$. 如果 $x_i \neq x_j, 1 \leqslant i < j \leqslant \ell$, 则环 C 称为单环.

例 12.2.2 考察例 12.2.1. 不难发现其中的环包括: (D), (H), (G), (H, D), (A, B), (F, C), (F, G, C), (A, F, C, B), (A, F, G, C, B), 等等.

设 $C := (s_1, \cdots, s_\ell)$ 为 $\mathcal{S}$ 中的一个环, 则环上的平均支付记为

$$P_a(C) = \frac{P(s_1) + \cdots + P(s_\ell)}{\ell}. \tag{12.2.7}$$

那么, 有如下结果[103].

命题 12.2.1 (i) 设 $S := (s_1, \cdots, s_T)$ 是系统 (12.1.1) 输入–状态转移图上的一条道路, 那么,

$$S = \bigcup_{i=1}^{N} C_i \cup R, \tag{12.2.8}$$

这里 C_i 为某些环, R 为不属于任何环的剩余, 并且 $|R| \leqslant k^{m+n}$.

(ii) 设 $S := (s_1, \cdots, s_T)$ 是一条道路, $\mathcal{C}$ 为环集合. 设 $C^* \in \mathcal{C}$, 使得

$$P_a(C^*) = \max\{P_a(C), \forall C \in \mathcal{C}\}.$$

那么

$$J(S) \leqslant P_a(C^*). \tag{12.2.9}$$

证明 (i) 从 S 中将环一个接一个地移去, 最后剩余的元素至多有 k^{m+n} 个元素. 因为 $|\mathcal{S}| = k^{m+n}$, 并且注意到 (12.2.7) 是按元素计算的, 即无须关心元素是否在图中 "挨着" 以形成环, 所以, 如果剩余元素多于 k^{m+n} 个元素, 则必有环.

(2) 将 S 分解为 $C_i, i = 1, \cdots, N$ 和 R. 设 $|C_i| = \ell_i$, 那么,

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^{T} P(S_t) &= \sum_{i=1}^{N}\left[\frac{\ell_1}{T}P_a(C_1) + \cdots + \frac{\ell_N}{T}P_a(C_N)\right] + \frac{P(r_1) + \cdots + P(r_q)}{T} \\
&\leqslant \frac{\sum\limits_{i=1}^{N}\ell_i}{T}P_a(C^*) + \frac{P(r_1) + \cdots + P(r_q)}{T} \\
&= \frac{T-q}{T}P_a(C^*) + \frac{P(r_1) + \cdots + P(r_q)}{T},
\end{aligned} \tag{12.2.10}$$

这里, $\{r_1, \cdots, r_q\} = R$, 因此 $q \leqslant k^{m+n}$. 因此,

$$\lim_{T\to\infty} \frac{T-q}{T}P_a(C^*) + \frac{P(r_1) + \cdots + P(r_q)}{T} = P_a(C^*),$$

结论显见. □

下面考察从 X_0 出发的轨线, 记 X_0 的可达集为

$$R(X_0) = \bigcup_{U_0 \in \mathcal{D}_k^m} R(S_0 = (U_0, X_0));$$

记这个可达集里的环为

$$\mathcal{C}_{X_0} = \{C \in \mathcal{C} \mid C \subset R(X_0)\}.$$

并且, $C^*_{X_0} \in \mathcal{C}_{X_0}$ 为最优环, 即

$$P_a(C^*_{X_0}) \geqslant P_a(C), \quad \forall C \in \mathcal{C}_{X_0}.$$

于是有如下推论.

推论 12.2.1　考察网络 (12.1.1) 在目标函数 (12.2.3) 下的最优控制.

(i) 最优控制是令轨线收敛于 $C^*_{X_0}$ 的控制, 并且, 最优控制下的目标函数值为

$$J_{\max} = P_a(C^*_{X_0}).$$

(ii) 如果 $C^* \subset R(X_0)$, 那么最优控制目标函数值为

$$J_{\max} = P_a(C^*).$$

(iii) 如果输入–状态转移图是强连通的, 则最优控制目标函数值为 $P_a(C^*)$, 它不依赖于初始点 X_0.

12.2.2　逻辑控制网络的拓扑结构

由前可知, 要找到平均值下的最优控制, 必须知道逻辑控制网络的拓扑结构.

本小节假定所论输入–状态转移图是全连通的. 否则, 可用 $R(x_0)$ 代替 $S = \Delta_k^{m+n}$.

考察 k 值逻辑控制网络 (12.1.1), 其代数状态空间表达式为 (12.2.1).

命题 12.2.2　$\delta_{k^{m+n}}(i,j)$ 为一条边, 当且仅当,

$$\mathrm{Col}_i(L) = \delta_{k^n}^{\ell}, \quad 这里\ \ell = j \pmod{k^n}. \tag{12.2.11}$$

证明　根据定义, 边 $\delta_{k^{m+n}}(i,j)$ 存在, 当且仅当, 存在控制 $u(t+1)$ 使得

$$u(t+1)L\delta_{k^{m+n}}^i = \delta_{k^{m+n}}^j. \tag{12.2.12}$$

因为 $L\delta_{k^{m+n}}^i = \mathrm{Col}_i(L)$, 由 (12.2.1) 可得

$$u(t+1)\,\mathrm{Col}_i(L) = \delta_{k^{m+n}}^j. \tag{12.2.13}$$

注意到 $\delta_{k^{m+n}}^j$ 可唯一分解为 $\delta_{k^m}^{\xi}\delta_{k^n}^{\ell}$, 这里 $j = (\xi-1)k^n + \ell$. 结论显见. □

例 12.2.3　考察布尔控制网络

$$x(t+1) = Lu(t)x(t), \tag{12.2.14}$$

这里, $u(t), x(t) \in \Delta$, 且

$$L = \delta_2[1,2,2,1].$$

注意到 $\delta_4^1 \sim (1,1)$, $\delta_4^2 \sim (1,0)$, $\delta_4^3 \sim (0,1)$, $\delta_4^4 \sim (0,0)$, 则可得输入–状态转移图 (图 12.2.2).

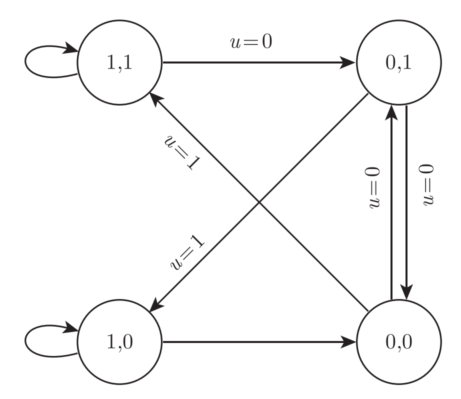


图 12.2.2 (12.2.14) 的输入–状态转移图

由图 12.2.2 可知, $(1,1)$ 和 $(1,0)$ 为不动点, 以下列举长度不超过 4 的环.

$(0,1)\to(0,0)$; $(0,1)\to(1,0)\to(0,0)$;
$(1,1)\to(0,1)\to(0,0)$; $(0,0)\to(1,1)\to(0,1)\to(1,0)$;
$(1,1)\to(1,1)\to(0,1)\to(0,0)$; $(1,0)\to(1,0)\to(0,0)\to(0,1)$.

在简单情况下输入–状态转移图的环可由图中直接找出, 但在 m 和 n 不是很小时, 从图上直接看出不是很容易, 因此需要寻找公式来计算它们.

由 (12.2.1) 出发, 可得

$$\begin{aligned} x(t+d) &= Lu(t+d-1)x(t+d-1) \\ &= Lu(t+d-1)Lu(t+d-2)\cdots Lu(t+1)Lu(t)x(t) \\ &= L(I_{k^m}\otimes L)u(t+d-1)u(t+d-2)Lu(t+d-3) \\ &\quad \ltimes Lu(t+d-4)\cdots Lu(t)x(t) \\ &:= L_d(\ltimes_{\ell=1}^{d}u(t+d-\ell))x(t), \end{aligned} \tag{12.2.15}$$

这里

$$L_d=\prod_{i=1}^{d}(I_{k^{(i-1)m}}\otimes L)\in\mathcal{L}_{k^n\times k^{dm+n}}. \tag{12.2.16}$$

在计算环之前我们需要一些记号.

(i) 设 $d\in\mathbb{Z}_+$, $\mathcal{P}(d)$ 记为 d 的真因子集合.

(ii) 设 $i,k,m\in\mathbb{Z}_+$, 那么,

$$\theta_k^m(d,i):=\{(\ell,j)\,|\,\exists\,\ell\in\mathcal{P}(d) \text{ 且 } (\ell,j) \text{ 满足 } \delta_{k^{dm}}^i=(\delta_{k^{\ell m}}^j)^{\frac{d}{\ell}}\}. \tag{12.2.17}$$

用一个简单例子解释一下上述记号.

例 12.2.4 (1) 设 $d=6$, 则 $\mathcal{P}(d)=\{1,2,3\}$.

(2) 如果 $m,k,d\in\mathbb{Z}_+$ 给定, 那么, 利用显见等式

$$\delta_{k^\alpha}^a\delta_{k^\beta}^b=\delta_{k^{\alpha+\beta}}^{(a-1)k^\beta+b}$$

可知, 对每一个 $\ell\in\mathcal{P}(d)$ 至多有一个 j 使得 $(\ell,j)\in\theta_k^m(i,d)$.

例如, 设 $m=k=2$, $d=6$.

(i) 如果 $i=1$, 那么

对 $\ell=1$, $\delta_{k^{dm}}^i=\delta_{2^{12}}^1=(\delta_{2^2}^1)^6=(\delta_{k^{\ell m}}^j)^{\frac{d}{\ell}}$, 故 $j=1$;

对 $\ell=2$, $\delta_{2^{12}}^1=(\delta_{2^4}^1)^3$, 故 $j=1$;

对 $\ell=3$, $\delta_{2^{12}}^1=(\delta_{2^6}^1)^2$, 故 $j=1$.

因此,

$$\theta_2^2(6,1)=\{(1,1),(2,1),(3,1)\}.$$

(ii) 如果 $i=2$, 那么, 对任何 $\ell\in\mathcal{P}(6)$ 和任何 $1\leqslant j\leqslant 2^{2\ell}$,

$$\delta_{2^{12}}^2\neq(\delta_{2^{2\ell}}^j)^{\frac{d}{\ell}},$$

因此, 不存在解 j, 即

$$\theta_2^2(6,2)=\varnothing.$$

(iii) 如果 $i=2^6+2$, 那么

对 $\ell=1$ 或 2, 不存在解 j;

对 $\ell=3$, $\delta_{2^{12}}^{2^6+2}=(\delta_{2^6}^2)^2$, 故 $j=2$.

因此,

$$\theta_2^2(6,2^6+2)=\{(3,2)\}.$$

下面我们将 $\theta_k^m(d,i)$ 简记为 $\theta(d,i)$, 设逻辑形式 k 和输入个数 m 预先给定.

记 $\mathrm{Blk}_i(L)$ 为 L 的第 i 个 $n\times n$ 块, 则有如下定理.

定理 12.2.1 考察网络 (12.1.1). 其输入–状态转移图中长度为 d 的环的个数可递推地计算如下:

$$N_d=\frac{1}{d}\sum_{i=1}^{k^{dm}}T(\mathrm{Blk}_i(L_d)),\tag{12.2.18}$$

这里, L_d 由式 (12.2.16) 确定, 且

$$T(\mathrm{Blk}_i(L_d))=\mathrm{tr}(\mathrm{Blk}_i(L_d))-\sum_{(\ell,j)\in\theta(d,i)}T(\mathrm{Blk}_j(L_\ell)).\tag{12.2.19}$$

证明 $\mathcal{S}$ 中的每一个环是状态空间的一个环与控制空间的一个环的乘积, 因此, 我们先寻找状态空间的环.

如果 $x(t)$ 在状态空间长度为 d 的环上, 根据 (12.2.15), 我们有

$$x(t) = L_d(\ltimes_{\ell=1}^{d} u(t+d-\ell))x(t).$$

如果 $u(t+d-1),\cdots,u(t)$ 确定, 记 $\ltimes_{\ell=1}^{d} u(t+d-\ell) = \delta_{k^{dm}}^{i}$, 那么

$$x(t) = \mathrm{Blk}_i(L_d)x(t).$$

如果 $x(t) = \delta_{k^n}^{j}$, 这表明 $\mathrm{Blk}_i(L_d)$ 的 (j,j) 位元素为 1. 因此, 状态空间长度为 d 的环在给定控制 $u(t+d-1),\cdots,u(t)$ 下为

$$\{x(t), Lu(t)x(t), L_2u(t+1)u(t)x(t),\cdots, L_du(t+d-1)\cdots u(t)x(t)\}.$$

因此, 将环与给定的 u 相乘, 我们得到一个在控制–状态乘积空间上长度为 d 的一个环. 因此, 长度为 d 的环, 包括有重复的环的个数为

$$N_d = \frac{1}{d}\sum_{i=1}^{k^{dm}} \mathrm{tr}(\mathrm{Blk}_i(L_d)).$$

显然, 如果 ℓ 是 d 的一个真因子, 并且 $x(t)$ 在控制 $\widetilde{u}(t+\ell-1)\cdots\widetilde{u}(t) = \delta_{k^{\ell m}}^{j}$ 下处于一个长度为 ℓ 的环上, 同时在控制 $u(t+d-1)\cdots u(t) = \delta_{k^{dm}}^{i}$ 下处于一个长度为 d 的环上, 那么我们得到同一个环, 当且仅当,

$$\delta_{k^{dm}}^{i} = (\delta_{k^{\ell m}}^{j})^{\frac{d}{\ell}}.$$

移除重复计算的环, 即得 (12.2.18). □

为检验 (12.2.19) 关于 d 归纳地定义了所有 $T(\mathrm{Blk}_i(L_d))$, 先令 $d=1$,

$$\theta(i,1) = \phi, \quad \forall\, i.$$

因此, 对于 $d=1$, $T(\mathrm{Blk}_i(L_d))$ 是定义好的. 因此, $d>1$ 可归纳地定义.

对于一个长度为 d 的环, 因为 $s(t) = \delta_{k^{m+n}}^{\ell}$ 可以被唯一地分解为 $u(t)x(t) = \delta_{k^m}^{i}\delta_{k^n}^{j}$, 这个环可以被表示成

$$C = \big((\delta_{k^m}^{i(t)}, \delta_{k^n}^{j(t)}), (\delta_{k^m}^{i(t+1)}, \delta_{k^n}^{j(t+1)}), \cdots, (\delta_{k^m}^{i(t+d-1)}, \delta_{k^n}^{j(t+d-1)})\big).$$

为记号简便, 将其记为

$$C = \delta_{k^m}\times\delta_{k^n}((i(t),j(t)),(i(t+1),j(t+1)),\cdots,(i(t+d-1),j(t+d-1))). \tag{12.2.20}$$

单环曾在定义 12.2.3 中定义过, 下面给出一个等价的定义.

定义 12.2.4 一个环 $C=\delta_{k^m}\times\delta_{k^n}((i(t),j(t)),(i(t+1),j(t+1)),\cdots,(i(t+d-1),j(t+d-1)))$ 称为单环, 如果它满足

$$i(\xi)\neq i(\ell),\quad t\leqslant\xi<\ell\leqslant t+d-1. \tag{12.2.21}$$

例 12.2.5 回忆例 12.2.3. 首先, 因为

$$L_1=L=\delta_2[1,2,2,1],$$

我们有 $\mathrm{tr}(\mathrm{Blk}_1(L_1))=2$, $\mathrm{tr}(\mathrm{Blk}_2(L_1))=0$. 因此, δ_2^1 及 δ_2^2 为控制 $u=\delta_2^1$ 的不动点. 从而可知 $N_1=1$ 且输入–状态转移图中的不动点是为

$$\delta_2\times\delta_2((1,1)),\quad \delta_2\times\delta_2((1,2)),$$

它们都是单环. 其次, 由于

$$L_2=L(I_2\otimes L)=\delta_2[1,2,2,1,2,1,1,2],$$

我们有

$$\mathrm{tr}(\mathrm{Blk}_1(L_2))=\mathrm{tr}(\mathrm{Blk}_4(L_2))=2,$$
$$\mathrm{tr}(\mathrm{Blk}_2(L_2))=\mathrm{tr}(\mathrm{Blk}_3(L_2))=0.$$

同时 $\delta_4^1=\delta_2^1\delta_2^1$, $\delta_4^4=\delta_2^2\delta_2^2$, 因此可得

$$T(\mathrm{Blk}_1(L_2))=\mathrm{tr}(\mathrm{Blk}_1(L_2))-T(\mathrm{Blk}_1(L_1))=0,$$
$$T(\mathrm{Blk}_4(L_2))=\mathrm{tr}(\mathrm{Blk}_4(L_2))-T(\mathrm{Blk}_2(L_1))=2,$$

且 $T(\mathrm{Blk}_2(L_2))=T(\mathrm{Blk}_3(L_2))=0$. 于是有 $N_2=1$. δ_2^1 及 δ_2^2 属于在控制 $u(t+1)u(t)=\delta_2^2\delta_2^2$ 下长度为 2 的环. 于是可得输入–状态转移图中长度为 2 的环如下:

$$\delta_2\times\delta_2((2,1),(2,2)),$$

它也是单环.

考察

$$L_3=L(I_2\otimes L)(I_4\otimes L)=\delta_2[1,2,2,1,2,1,1,2,1,2,2,1].$$

因为

$$\mathrm{tr}(\mathrm{Blk}_1(L_3))=\mathrm{tr}(\mathrm{Blk}_4(L_3))=\mathrm{tr}(\mathrm{Blk}_6(L_3))=\mathrm{tr}(\mathrm{Blk}_7(L_3))=2$$

可知

$$T(\mathrm{Blk}_4(L_3)) = T(\mathrm{Blk}_6(L_3)) = T(\mathrm{Blk}_7(L_3)) = 2$$

且

$$T(\mathrm{Blk}_i(L_3)) = 0, \quad i = 1, 2, 3, 5, 8,$$

于是有 $N_3 = 2$. δ_2^1 及 δ_2^2 在控制

$$\begin{aligned} & u(t+2)u(t+1)u(t) = \delta_8^4 = \delta_2^1\delta_2^2\delta_2^2 \\ & \delta_8^6 = \delta_2^2\delta_2^1\delta_2^2, \quad \delta_8^7 = \delta_2^2\delta_2^2\delta_2^1 \end{aligned}$$

下的长度为 3 的环上, 由此可得输入–状态转移图中长度为 3 的环如下:

$$\delta_2 \times \delta_2((1,1),(2,1),(2,2)), \quad \delta_2 \times \delta_2((2,1),(1,2),(2,2)).$$

最后, 由于

$$\begin{aligned} L_4 &= L(I_2 \otimes L)(I_4 \otimes L)(I_8 \otimes L) \\ &= \delta_2[1,2,2,1,2,1,1,2,2,1,1,2,1,2,2,1,2,1,1,2,1,2,2,1,1,2,2,1,2,1,1,2], \end{aligned}$$

我们有

$$\mathrm{tr}(\mathrm{Blk}_i(L_4)) = 2, \quad i = 1, 4, 6, 7, 10, 11, 13, 16.$$

于是

$$T(\mathrm{Blk}_i(L_4)) = \begin{cases} 2, & i = 4, 6, 7, 10, 11, 13, \\ 0, & \text{其他}. \end{cases}$$

综上可知 $N_4 = 3$, 并且, δ_2^1 和 δ_2^2 处于在控制

$$\begin{aligned} & u(t+3)u(t+2)u(t+1)u(t) = \delta_{16}^4 = \delta_2^1\delta_2^1\delta_2^2\delta_2^2, \\ & \delta_{16}^6 = \delta_2^1\delta_2^2\delta_2^1\delta_2^2, \\ & \delta_{16}^7 = \delta_2^1\delta_2^2\delta_2^2\delta_2^1, \\ & \delta_{16}^{10} = \delta_2^2\delta_2^1\delta_2^1\delta_2^2, \\ & \delta_{16}^{11} = \delta_2^2\delta_2^1\delta_2^2\delta_2^1, \\ & \delta_{16}^{13} = \delta_2^2\delta_2^2\delta_2^1\delta_2^1 \end{aligned}$$

下的长度为 4 的环上, 由此可得输入–状态转移图中长度为 4 的环如下:

$$\begin{aligned} & \delta_2 \times \delta_2((1,1),(2,1),(1,2),(2,2)), \\ & \delta_2 \times \delta_2((1,2),(1,2),(2,2),(2,1)), \\ & \delta_2 \times \delta_2((1,1),(1,1),(2,1),(2,2)). \end{aligned}$$

这个结果与例 12.2.3 中从输入–状态转移图看出的结果一致.

12.2.3 最优控制设计

考察逻辑控制系统的最优控制和最优轨道, 推论 12.2.1 将文献 [103] 中关于单变量的相应结果推广到多变量 k 值逻辑网络. 但如果将这些问题转到输入–状态转移图上来考虑, 就会发现它们本质上无大差别, 因此, 证明也是类似的.

推论 12.2.1 说明, 最优控制可在一条收敛于“最优” 环 C^* 的轨线上实现. 后面将证明, 存在一个逻辑矩阵 G^*, 称为最优控制矩阵, 使得最优控制可表示为

$$u^*(t+1) = G^*u^*(t)x^*(t). \tag{12.2.22}$$

命题 12.2.3 极限值

$$J(u^*) := \lim_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}P(x^*(t), u^*(t)) \tag{12.2.23}$$

总存在.

证明 考察系统 (12.1.1), 根据引理 12.2.1, 一条最优轨道将收敛到一个吸引子. $J(u^*)$ 就是在这个吸引子上的平均值. □

命题 12.2.4 对任何一个环 C, 存在一个单环 $C_s\subset C$, 使得

$$P_a(C_s) \geqslant P_a(C). \tag{12.2.24}$$

证明 记 $C=\delta_{k^m}\times\delta_{k^n}((i(t),j(t)),(i(t+1),j(t+1)),\cdots,(i(t+d-1),j(t+d-1)))$ 为任意一环. 如果它是单环, 结论显见. 现在假定 $\delta_{k^n}^{j(\xi)}=\delta_{k^n}^{j(\ell)}$, $\xi<\ell$, 且 $C_1=\delta_{k^m}\times\delta_{k^n}((i(\xi),j(\xi)),\cdots,(i(\ell-1),j(\ell-1)))$ 为一单环. 如果 $P_a(C_1)\geqslant P_a(C)$, 结论显见.

否则, 我们移去 C_1, 那么, 剩余结点形成一个新的环 C_1', 因为 $L\delta_{k^m}^{i(\xi-1)}\delta_{k^n}^{j(\xi-1)}=\delta_{k^n}^{i(\xi)}=\delta_{k^n}^{i(\ell)}$, 现在 $P_a(C_1')>P_a(C)$. 如果 C_1' 是单环, 结论显见.

否则, 我们可以在其中找出一个单环 C_2. 现在, 或者 (12.2.24) 成立, 或者将其移去. 继续这个过程, 最后我们总能找到一个单环 C_s 满足 (12.2.24). □

根据 (12.2.15), 在第 d 步, 初态 x_0 可以到达

$$R_d(x_0)=\{u(d)L_d\ltimes_{\ell=1}^{d}u(d-\ell)x_0\,|\,\forall\, u(\ell)\in\Delta_{k^m}, 0\leqslant\ell\leqslant d\},$$

如果 $x_0=\delta_{k^n}^{j(0)}$, 则

$$R_d(x_0)=\{u(d)\,\mathrm{Col}_\ell(L_d)\,|\,\forall\, u(d)\in\Delta_{k^m}, \ell=j(0)\ (\mathrm{mod}\ k^n)\}.$$

如果 $\delta_{k^m}^i\delta_{k^n}^j$ 是从 x_0 出发并在 d 步可到达, $d \geqslant k^n$, 则轨线 $\delta_{k^m}^i\delta_{k^n}^j$ 必然至少通过某点两次. 类似于命题 12.2.4 的证明, 我们最后可知, 状态 $\delta_{k^m}^i\delta_{k^n}^j$ 是从 x_0 出发, 在 d' 步可达, 这里 $d' < k^n$. 因此,

$$R(x_0) = \bigcup_{d=0}^{k^n-1} R_d(x_0). \tag{12.2.25}$$

根据以上讨论, 我们可以从在 $R(x_0)$ 中的单环中找到最优环 C^*, 记从初始状态到 C^* 的最短路径为

$$\left(\delta_{k^m}^{i(0)}\delta_{k^n}^{j(0)}, \delta_{k^m}^{i(1)}\delta_{k^n}^{j(1)}, \cdots, \delta_{k^m}^{i(T_0-1)}\delta_{k^n}^{j(T_0-1)}\right), \tag{12.2.26}$$

这里,

$$C^* = \delta_{k^m} \times \delta_{k^n}((i(T_0), j(T_0)), \cdots, (i(T_0+d-1), j(T_0+d-1))).$$

我们称 (12.2.26) 为最优轨道.

下面证明, 存在最优控制矩阵 G^*, 使最优控制可表示为 (12.2.22).

定理 12.2.2　*考察 k 值逻辑控制网络 (12.1.1) 及目标函数 (12.2.3). 设最优轨道为 (12.2.26), 且最优控制为 $u^*(t)$. 则存在逻辑矩阵 $G^* \in \mathcal{L}_{k^n \times k^{n+m}}$, 满足*

$$\begin{cases} x^*(t+1) = Lu^*(t)x^*(t), \\ u^*(t+1) = G^*u^*(t)x^*(t). \end{cases} \tag{12.2.27}$$

证明　根据命题 12.2.4, 可以从单环中找出最优环. 由于一个单环的长度不超过 k^n, 设一条轨道的初始点为 $\delta_{k^n}^{j(0)}$, 可以找出从该初始点出发可到达的长度不超过 k^n 的环, 然后可找到最优环. 易知 $T_0 + d \leqslant k^{m+n}$, 故可找到最优控制矩阵 G^* 的 $T_0 + d$ 列如下:

$$\mathrm{Col}_s(G^*) = \begin{cases} \delta_{k^m}^{i(\ell+1)}, & s = (i(\ell)-1)k^n + j(\ell), \quad \ell \leqslant T_0+d-2, \\ \delta_{k^m}^{i(T_0)}, & s = (i(T_0+d-1)-1)k^n + j(T_0+d-1), \end{cases} \tag{12.2.28}$$

G^* 的其他列 ($\mathrm{Col}(G^*) \subset \Delta_{k^m}$) 可任选. 因此, G^* 可构造. □

例 12.2.6　回忆例 12.2.3 及例 12.2.5. 设

$$P(u(t), x(t)) = u^{\mathrm{T}}(t) \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} x(t).$$

假定初态 $x_0 = \delta_2^2$, 从例 12.2.5 的结果可知 $C^* = \delta_2 \times \delta_2((2,1),(2,2))$ 是最优环. 选 $u(0) = \delta_2^2$, 则最优环及从 δ_2^2 到达最优环的最小路径为

$$\delta_2 \times \delta_2((2,2),(2,1)).$$

因此,

$$G^* = \delta_2[i, j, 2, 2],$$

这里, $i, j \in \{1, 2\}$ 可以任选.

例 12.2.7　考察一个无限重复的博弈. 这里有两个玩家: 玩家 1 和玩家 2. 每个玩家都有三种策略可选, 为 $\{L, M, R\}$. 支付双矩阵见表 12.2.1.

表 12.2.1　例 12.2.7 的支付双矩阵

P_1	P_2		
	L	M	R
L	$3, 3$	$0, 4$	$9, 2$
M	$4, 0$	$4, 4$	$5, 3$
R	$2, 9$	$3, 5$	$6, 6$

不难验证 (M, M) (即玩家 1 选策略 M, 玩家 2 也选策略 M) 是唯一的一次博弈的纳什均衡点. 但显然 (R, R) 是比 (M, M) 更好的策略. 现在假定玩家 1 是人, 玩家 2 是机器, 它按以下固定方式选择策略. ⓐ 第一次选择: R; ⓑ 第 t 次选择: 如果在 $(t-1)$ 次双方策略为 (R, R), 则在第 t 次选 R, 否则选 M. 这种策略称为触发策略[52].

记 $L \sim \delta_3^1$, $M \sim \delta_3^2$, $R \sim \delta_3^3$. 玩家 1 的策略为 $u(t)$, 玩家 2 的策略为 $x(t)$, 则上述博弈模型变为一个三值逻辑动态系统

$$x(t+1) = Lu(t)x(t), \tag{12.2.29}$$

这里

$$L = \delta_3[2, 2, 2, 2, 2, 2, 2, 2, 3],$$

$x(t) \in \varDelta_3$ 为状态, 是玩家 2 在 t 时刻的策略, $u(t) \in \varDelta_3$ 为控制, 是玩家 1 在 t 时刻的策略.

文献 [52] 指出, 对于无限重复博弈, 记其支付函数为

$$(1-\delta)\sum_{t=1}^{\infty} \delta^{t-1}\pi_t,$$

这里 π_t 是在 t 时刻的支付, δ 是折扣因子, 当 δ 足够接近 1 时, 触发策略是一个纳什均衡.

忽略折扣, 我们考虑平均支付, 则玩家 1 与玩家 2 的支付分别为

$$J_1 = \overline{\lim_{T\to\infty}} \frac{1}{T}\sum_{t=1}^{T} P_1(x(t), u(t)),$$

$$J_2 = \overline{\lim_{T\to\infty}} \frac{1}{T}\sum_{t=1}^{T} P_2(x(t),u(t)),$$

这里

$$P_1(x(t),u(t)) = u^{\mathrm{T}}(t)\begin{bmatrix}3 & 0 & 9\\ 4 & 4 & 5\\ 2 & 3 & 6\end{bmatrix} x(t),$$

$$P_2(x(t),u(t)) = u^{\mathrm{T}}(t)\begin{bmatrix}3 & 4 & 2\\ 0 & 4 & 3\\ 9 & 5 & 6\end{bmatrix} x(t).$$

一个自然的问题是: 此时触发策略是否还是一个纳什均衡? 因为玩家 2 已经采用了触发策略, 我们要找到玩家 1 的最优策略, 看是否为触发策略? 于是, 问题转化为三值逻辑网络 (12.2.29) 关于 J_1 的最优控制问题.

现在计算环,

$$L_1 = L = \delta_3[2,2,2,2,2,2,2,2,3],$$

于是有

$$\begin{aligned}&\mathrm{tr}(\mathrm{Blk}_1(L_1)) = 1,\\ &\mathrm{tr}(\mathrm{Blk}_2(L_1)) = 1,\\ &\mathrm{tr}(\mathrm{Blk}_3(L_1)) = 2,\end{aligned}$$

$N_1 = 4$. δ_3^2 是控制 $u = \delta_3^i, i = 1,2,3$ 下的不动点, δ_3^3 是在 $u = \delta_3^3$ 下的不动点. 于是系统 (12.2.29) 的不动点为

$$\delta_3 \times \delta_3((1,2)),\ \delta_3 \times \delta_3((2,2)),\ \delta_3 \times \delta_3((3,2)) \text{ 及 } \delta_3 \times \delta_3((3,3)).$$

$$L_2 = L(I_3 \otimes L) = \delta_3[2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,3],$$

$$\mathrm{tr}(\mathrm{Blk}_i(L_2)) = 1, \quad i = 1,\cdots,8, \quad \mathrm{tr}(\mathrm{Blk}_9(L_2)) = 2.$$

对于控制 $u(t+1)u(t) = \delta_9^1 = \delta_3^1\delta_3^1$,

$$T(\mathrm{Blk}_1(L_2)) = \mathrm{tr}(\mathrm{Blk}_1(L_2)) - \mathrm{tr}(\mathrm{Blk}_1(L_1)) = 0.$$

类似地, 可得

$$\begin{aligned}&T(\mathrm{Blk}_1(L_2)) = T(\mathrm{Blk}_5(L_2)) = T(\mathrm{Blk}_9(L_2)) = 0,\\ &T(\mathrm{Blk}_i(L_2)) = 1, \quad i = 2,3,4,6,7,8.\end{aligned}$$

由此可知 $N_2=3$. δ_3^2 是在控制 $u(t+1)u(t)=\delta_9^i, i=2,3,4,6,7,8$ 下长度为 2 的环. 因此, 我们可得长度为 2 的环如下:

$$\delta_3\times\delta_3((1,2),(2,2)),\quad \delta_3\times\delta_3((1,2),(3,2)),\quad \delta_3\times\delta_3((2,2),(3,2)).$$

$$L_3=L_x(I_3\otimes L_x)(I_9\otimes L_x)=\delta_{81}[\underbrace{2,\cdots,2}_{80},3].$$

根据 (12.2.19) 可得

$$\begin{aligned}&T(\mathrm{Blk}_i(L_3))=1,\quad i=2,\cdots,13,15,\cdots,26,\\&T(\mathrm{Blk}_i(L_3))=0,\quad i=1,14,27,\end{aligned}$$

并且 $N_3=8$. δ_3^2 是在控制 $u(t+2)u(t+1)u(t)=\delta_{27}^i, i=2,\cdots,13,15,\cdots,26$ 下长度为 3 的环. 因此, 我们可得长度为 3 的环如下:

$$\begin{aligned}&\delta_3\times\delta_3((1,2),(1,2),(2,2)),\quad \delta_3\times\delta_3((1,2),(3,2),(2,2)),\\&\delta_3\times\delta_3((1,2),(1,2),(3,2)),\quad \delta_3\times\delta_3((1,2),(3,2),(3,2)),\\&\delta_3\times\delta_3((1,2),(2,2),(2,2)),\quad \delta_3\times\delta_3((2,2),(2,2),(3,2)),\\&\delta_3\times\delta_3((1,2),(2,2),(3,2)),\quad \delta_3\times\delta_3((2,2),(3,2),(3,2)).\end{aligned}$$

虽然这里有许多长度大于或等于 4 的环, 但我们已经证明, 只要在长度小于或等于 3 的环中寻找最优环即可.

根据激发策略, 从初始点 $x_0=\delta_3^3$ 出发, 可达集为

$$R(x_0)=\{\delta_3^1\delta_3^2,\delta_3^2\delta_3^2,\delta_3^3\delta_3^2,\delta_3^1\delta_3^3,\delta_3^2\delta_3^3,\delta_3^1\delta_3^3\}.$$

利用前面的结果可知, 所有在 $R(x_0)$ 中的单环有

$$\begin{aligned}&\delta_3\times\delta_3((1,2)),\quad \delta_3\times\delta_3((2,2)),\\&\delta_3\times\delta_3((3,2)),\quad \delta_3\times\delta_3((3,3)),\end{aligned}$$

其中

$$C^*=\delta_3\times\delta_3((3,3))$$

为最优环. 选

$$u^*(0)=\delta_3^3,$$

则

$$G^*=\delta_3[*,*,*,*,*,*,*,*,3],$$

这里, 前 8 列可任选, 例如, 可选

$$G^* = \delta_3[2, 2, 2, 2, 2, 2, 2, 2, 3],$$

这就导致触发策略. 由此可见, 如果玩家 2 取触发策略, 那么, 玩家 1 也取触发策略是最优策略. 由于支付双矩阵是对称的, 如果玩家 1 取触发策略, 那么, 玩家 2 取触发策略也是最优策略. 因此, "触发策略–触发策略" 是这个博弈的纳什均衡点.

12.3　概率逻辑网络的优化控制

12.3.1　有限时间最优控制

考察 k 值逻辑控制网络 (12.1.1), 设其为概率逻辑控制网络, 即, F_i 具有 s_i 个可能形式 F_i^j, $j = 1, \cdots, s_i$, 且

$$\mathbb{P}(F_i = F_i^j) = p_i^j, \quad j = 1, \cdots, s_i, \quad i = 1, \cdots, n, \tag{12.3.1}$$

这里,

$$\sum_{j=1}^{s_i} p_i^j = 1, \quad i = 1, \cdots, n.$$

设目标函数为

$$J = \sum_{t=0}^{T} \lambda^t C(X_1(t), \cdots, X_n(t), U_1(t), \cdots, U_m(t)), \tag{12.3.2}$$

这里, $0 < \lambda < 1$ 为折扣因子, $C : \mathcal{D}_k^{n+m} \to \mathbb{R}$ 称为伪 k 值逻辑函数, $0 < T \leqslant \infty$.

现在, 系统 (12.1.1) 共有 $s = \prod_{i=1}^{n} s_i$ 个可能模式, 记作

$$F^\ell = \left(F_1^{\ell_1}, F_2^{\ell_2}, \cdots, F_n^{\ell_n}\right), \quad 1 \leqslant \ell_i \leqslant s_i, \quad i = 1, 2, \cdots, n, \tag{12.3.3}$$

这里,

$$\mathbb{P}(F = F^\ell) = \prod_{i=1}^{n} p_i^{\ell_i} := p^\ell.$$

设 F^ℓ 的结构矩阵为 M^ℓ, 则网络 (12.1.1) 依概率分布 (12.3.1) 形成的系统的代数状态空间表示为

$$x(t+1) = Mu(t)x(t), \tag{12.3.4}$$

这里,

$$
\begin{aligned}
M &= \sum_{\ell} p^{\ell} M^{\ell} \\
&= \sum_{\ell_1=1}^{s_1} \sum_{\ell_2=1}^{s_2} \cdots \sum_{\ell_n=1}^{s_n} \prod_{i=1}^{n} p_i^{\ell_i} M^{(\ell_1,\ell_2,\cdots\ell_n)}.
\end{aligned} \tag{12.3.5}
$$

注 12.3.1　实际上, $Mu(t)$ 是依赖于控制的 (列) 马尔可夫转移矩阵. 记

$$
M := A(u) = (a_{i,j}(u)).
$$

则

$$
\mathbb{P}(X(t+1) = i \mid X(t) = j, U(t) = U) = a_{i,j}(U). \tag{12.3.6}
$$

定义 12.3.1　考察概率 k 值逻辑控制网络 (12.1.1)—(12.3.1), 最优控制问题指对每个给定初始状态 x_0, 找到最优控制, 使

$$
J(x(0)) = \mathbb{E}\left[\sum_{t=0}^{T} \lambda^t C(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)) \middle| x(0)\right] \tag{12.3.7}
$$

达到最大.

本节讨论有限时长的情况, 即令 $T = N < \infty$. 这时, 可利用动态规划求最优解. 记

$$
C_t = \lambda^t C(u(t), x(t)).
$$

由动态规划原理, 可从后往前寻找最优解, 即从 $t = N$ 到 $t = 0$ 逐次寻优.

记

$$
r = k^m, \quad \kappa = k^n.
$$

下面这个命题直接来自动态规划原理[15,39]:

命题 12.3.1　设 $J^*(x(0))$ 为 (12.3.7) 的最优值, 那么,

$$
J^*(x(0)) = J_0(x(0)), \tag{12.3.8}
$$

这里, 函数 J_0 有动态规划最后一步给出的最优解. 动态规划算法可详述如下: 令

$$
C_t := \lambda^t C(u(t), x(t)), \tag{12.3.9}
$$

那么,

$$
J_N(x(N)) = \max_{u(N)\in\Delta_r} C_N(u(N), x(N)), \tag{12.3.10}
$$

以及

$$J_t(x(t)) = \max_{u(t)\in\Delta_r} \mathbb{E}\left[C_t(u(t),x(t)) + J_{t+1}(x(t+1))\,|\, x(t),u(t)\right],$$
$$t=0,1,\cdots,N-1. \tag{12.3.11}$$

注意到 (12.3.11) 右手边的期望是基于 $x(t)$ 和 $u(t)$ 的, 利用 (12.3.6) 可得

$$\mathbb{E}\left[J_{t+1}(x(t+1))|x(t),u(t)\right] = \sum_{i=1}^{\kappa} a_{i,x(t)}(u(t))J_{t+1}(i). \tag{12.3.12}$$

考察目标函数 (12.3.7). 记 $C(\delta_r^i,\delta_k^j) =: \varphi_{i,j}$, $i=1,\cdots,r$; $j=1,\cdots,\kappa$, 则可定义一个矩阵, 称为支付矩阵, 如下:

$$\Phi := (\varphi_{i,j}) \in \mathcal{M}_{r\times\kappa}. \tag{12.3.13}$$

则得

$$C_t = \lambda^t C(u(t),x(t)) = \lambda^t u^{\mathrm{T}}(t)\Phi x(t). \tag{12.3.14}$$

于是优化问题的动态规划解可表示如下:

$$\begin{cases} J_N(x(N)) = \max\limits_{u(N)\in\Delta_r} \lambda^N u^{\mathrm{T}}(N)\Phi x(N), \\ J_t(x(t)) = \max\limits_{u(t)\in\Delta_r} \left[\lambda^t u^{\mathrm{T}}(t)\Phi x(t) + \sum\limits_{i=1}^{\kappa} a_{i,x(t)}(u(t))J_{t+1}(i)\right], \\ \qquad t=N-1,N-2,\cdots,1,0. \end{cases} \tag{12.3.15}$$

将 $J_t(x(t))$ 根据不同的 $x(t)\in\{\delta_k^i|i=1,\cdots,\kappa\}$ 排成向量形式:

$$J_t := \begin{bmatrix} J_t(\delta_\kappa^1) \\ J_t(\delta_\kappa^2) \\ \vdots \\ J_t(\delta_\kappa^\kappa) \end{bmatrix},$$

那么, (12.3.16) 的第一式变为

$$J_N(x(N)) = J_N x(N), \tag{12.3.16}$$

这里

$$J_N = \lambda^N \begin{bmatrix} \max\limits_{u(N)\in\Delta_r} u^{\mathrm{T}}(N)\,\mathrm{Col}_1(\Phi) \\ \max\limits_{u(N)\in\Delta_r} u^{\mathrm{T}}(N)\,\mathrm{Col}_2(\Phi) \\ \vdots \\ \max\limits_{u(N)\in\Delta_r} u^{\mathrm{T}}(N)\,\mathrm{Col}_\kappa(\Phi) \end{bmatrix}. \tag{12.3.17}$$

注意到 $u(N)\in\varDelta_r$, 记

$$\xi^i(N) := \mathrm{Col}_i(\Phi),$$

那么对 $J_N(x(N))\big|_{x(N)=\delta_\kappa^i}$ 的最优控制为

$$u_i^*(N) = \delta_r^{j^*}, \tag{12.3.18}$$

这里

$$j^* = \arg\max_j \xi_j^i(N), \tag{12.3.19}$$

这里 $\xi_j^i(N)$ 是 $\xi^i(N)$ 的第 j 个分量.

类似地, (12.3.16) 的第二式可表示为

$$J_t(x(t)) = J_t x(t), \tag{12.3.20}$$

这里

$$J_t = \begin{bmatrix} \max\limits_{u(t)\in\varDelta_r} \left\{\lambda^t u^{\mathrm{T}}(t)\,\mathrm{Col}_1(\Phi) + \mathrm{Row}_1[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}]J_{t+1}\right\} \\ \max\limits_{u(t)\in\varDelta_r} \left\{\lambda^t u^{\mathrm{T}}(t)\,\mathrm{Col}_2(\Phi) + \mathrm{Row}_2[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}]J_{t+1}\right\} \\ \vdots \\ \max\limits_{u(t)\in\varDelta_r} \left\{\lambda^t u^{\mathrm{T}}(t)\,\mathrm{Col}_\kappa(\Phi) + \mathrm{Row}_\kappa[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}]J_{t+1}\right\} \end{bmatrix}. \tag{12.3.21}$$

注意到 $x^{\mathrm{T}}u^{\mathrm{T}}(L^*)^{\mathrm{T}} = u^{\mathrm{T}}x^{\mathrm{T}}W_{[r,\kappa]}(M)^{\mathrm{T}}$, 定义

$$B := W_{[r,\kappa]}[(L^*)^{\mathrm{T}}] =: \begin{bmatrix} B_1 \\ \vdots \\ B_\kappa \end{bmatrix}, \tag{12.3.22}$$

这里 $B_i\in\mathcal{M}_{r\times\kappa}$, $i=1,\cdots,\kappa$. 于是有

$$\mathrm{Row}_i[u^{\mathrm{T}}(t)(L^*)^{\mathrm{T}}] = u(t)^{\mathrm{T}}B_i, \quad i=1,\cdots,\kappa. \tag{12.3.23}$$

利用 (12.3.23), (12.3.21) 可写成

$$J_t = \begin{bmatrix} \max\limits_{u(t)\in\varDelta_r} u^{\mathrm{T}}(t)\,[\lambda^t\,\mathrm{Col}_1(\Phi) + B_1J_{t+1}] \\ \max\limits_{u(t)\in\varDelta_r} u^{\mathrm{T}}(t)\,[\lambda^t\,\mathrm{Col}_2(\Phi) + B_2J_{t+1}] \\ \vdots \\ \max\limits_{u(t)\in\varDelta_r} u^{\mathrm{T}}(t)\,[\lambda^t\,\mathrm{Col}_\kappa(\Phi) + B_\kappa J_{t+1}] \end{bmatrix}. \tag{12.3.24}$$

由于 $u(t) \in \Delta_r$, 令

$$\xi^i(t) = \lambda^t \operatorname{Col}_i(\Phi) + B_i J_{t+1} \in \mathbb{R}^r,$$

那么, 对于

$$J_t(x(t))\big|_{x(t)=\delta_\kappa^i}$$

的最优控制为

$$u_i^*(t) = \delta_r^{j^*}, \tag{12.3.25}$$

这里

$$j^* = \arg\max_j \xi_j^i(t). \tag{12.3.26}$$

公式 (12.3.18) 与 (12.3.25) 给出了完整的最优控制.

12.3.2 无限时间-基于预测控制的优化

考察 k 值逻辑控制网络 (12.1.1)—(12.3.1) 在如下无穷长时间上的优化控制问题:

$$J(x(0)) := \mathbb{E}\left[\sum_{t=0}^{\infty} \lambda^t C(u(t), x(t)) \middle| x(0)\right]. \tag{12.3.27}$$

这里介绍的是文献 [33] 中提出的用预测控制进行优化控制设计. 预测控制在连续系统控制中有着广泛应用[77]. 在逻辑系统中, 其基本想法是: 对 $s = 0, 1, 2, \cdots$ 依次优化

$$J_s^{s+\ell-1}(x(s), u(s), u(s+1), \cdots, u(s+\ell-1)) := \mathbb{E}\left[\sum_{t=s}^{s+\ell-1} \lambda^t C(u(t), x(t)) \middle| x(s)\right], \tag{12.3.28}$$

对每个 s 得到相应的最优控制 $u^*(x)$, 从而得到最优控制序列 $u^*(0), u^*(1), \cdots$.

观察 (12.3.28), 不难发现最优控制是一个时不变的状态反馈控制, 即有如下结论.

命题 12.3.2 对无穷时间优化问题, 最优控制可表示为

$$u^*(t) = \Psi x(t), \tag{12.3.29}$$

这里 $\Psi \in \mathcal{L}_{r \times k}$ 是一个逻辑矩阵.

定义

$$d := \min_{x\in\Delta_k} \min_{u_i\neq u_j\in\Delta_r} \left|c(x,u_i)-c(x,u_j)\right|;$$

$$M := \max_{u\in\Delta_r, x\in\Delta_k} \left|c(u,x)\right| < \infty.$$

下面给出一个假定条件.

条件 A-1: 如果

$$\begin{aligned}&J_s^{s+\ell-1}\left(x(s),u_s^{\ell-1}(s),u_s^{\ell-1}(s+1),\cdots,u_s^{\ell-1}(s+\ell-1)\right)\\&-J_s^{s+\ell-1}\left(x(s),u_s^{\infty}(s),u_s^{\infty}(s+1),\cdots,u_s^{\infty}(s+\ell-1)\right)\in[0,d),\end{aligned}\tag{12.3.30}$$

则

$$\begin{aligned}&J_s^{s+\ell-1}\left(a_{x_{s+1},x_s}(u_s^{\ell-1})x(s),u_s^{\ell-1}(s+1),\cdots,u_s^{\ell-1}(s+\ell-1)\right)\\&-J_s^{s+\ell-1}\left(a_{x_{s+1},x_s}(u_s^{\infty})x(s),u_s^{\infty}(s+1),\cdots,u_s^{\infty}(s+\ell-1)\right)\in[0,d),\\&\qquad s=0,1,2,\cdots.\end{aligned}\tag{12.3.31}$$

定理 12.3.1　设 $d>0$ 且条件 A-1 成立, 则当滤波长度

$$\ell > \log_{\lambda}\frac{(1-\lambda)d}{2M}\tag{12.3.32}$$

时, 由预测控制得到的最优控制 $u^*(0),u^*(1),\cdots$ 就是无穷时间上的最优控制. ①

证明　利用 (12.3.32) 可得

$$\left(\lambda^{\ell}+\lambda^{\ell+1}+\lambda^{\ell+2}+\cdots\right)M=\frac{\lambda^{\ell}}{1-\lambda}M<\frac{d}{2}.\tag{12.3.33}$$

设

$$u_0^{\ell-1} := (u_0^{\ell-1}(0),u_0^{\ell-1}(1),\cdots,u_0^{\ell-1}(\ell-1))$$

是由 (12.3.28) 定义的

$$J_0^{\ell-1}(x(0),u_0^{\ell-1})$$

的 (有限时间) 最优解, $\{u_0^{\infty}(0),u_0^{\infty}(1),\cdots\}$ 是

$$\max_{u(0),u(1),\cdots} J_0^{\infty}$$

的 (无限时间) 最优解. 我们要证明, 如果 ℓ 满足 (12.3.32), 则

$$0 \leqslant J_0^{\ell-1}\left(x(0),u_0^{\ell-1}(0),\cdots,u_0^{\ell-1}(\ell-1)\right)$$

① 定理 12.3.1 来自文献 [33]. 但原文不假定条件 A-1, 故结论有问题. 文献 [123] 给出了修正, 但其估计较保守. 本书首作者感谢文献 [123] 作者指出 [33] 的错误.

$$-J_0^{\ell-1}(x(0),u_0^\infty(0),\cdots,u_0^\infty(\ell-1))<d. \tag{12.3.34}$$

实际上 “$\geqslant 0$” 来自 $\left\{u_0^{\ell-1}(i)|i=0,\cdots,\ell-1\right\}$ 的最优性, 因此, 只要证 “$< d$” 就行了. 首先, 由 M 的定义可知

$$|EC(x(t),u(t))|\leqslant M. \tag{12.3.35}$$

设 (12.3.34) 右半不等式不成立, 那么

$$\begin{aligned}
&J_0^\infty\left(x(0),u_0^{\ell-1}(0),\cdots,u_0^{\ell-1}(\ell-1),u_0^\infty(\ell),u_0^\infty(\ell+1),\cdots\right)\\
&-J_0^\infty(x(0),u_0^\infty(0),\cdots,u_0^\infty(\ell-1),\cdots)\\
>&\left[J_0^{\ell-1}\left(x(0),u_0^{\ell-1}(0),\cdots,u_0^{\ell-1}(\ell-1)\right)-\frac{d}{2}\right]\\
&-\left[J_0^{\ell-1}(x(0),u_0^\infty(0),\cdots,u_0^\infty(\ell-1))+\frac{d}{2}\right]\\
\geqslant& d-2\frac{d}{2}=0,
\end{aligned} \tag{12.3.36}$$

这与 $\{u_0^\infty(i)|i=0,1,\cdots\}$ 的最优性矛盾.

其次, 假定 $u_0^{\ell-1}(0)\neq u_0^\infty(0)$. 那么, 因为 ℓ 满足 (12.3.32), 则有 (12.3.34). 根据动态规划基本原理, $\{u_0^\infty(1),u_0^\infty(2),\cdots\}$ 也是 J_1^∞ 的最优控制, 并且 $u_0^{\ell-1}(1)$, $u_0^{\ell-1}(2)$, $\cdots$, $u_0^{\ell-1}(\ell-1)$ 也是 $J_1^{\ell-1}$ 的最优控制. 利用条件条件 A-1 可知

$$\begin{aligned}
0\leqslant& J_1^{\ell-1}\left(a_{x(1),x(0)}(u_0^{\ell-1}(0))x(0),u_0^{\ell-1}(1),\cdots,u_0^{\ell-1}(\ell-1)\right)\\
&-J_1^{\ell-1}\left(a_{x(1),x(0)}(u_0^\infty(0))x(0),u_0^\infty(1),\cdots,u_0^\infty(\ell-1)\right)\\
<& d.
\end{aligned} \tag{12.3.37}$$

将 (12.3.37) 代入 (12.3.34) 可得

$$\begin{aligned}
0\leqslant&\left[c(x(0),u_0^{\ell-1}(0))+J_1^{\ell-1}\left(a_{x(1),x(0)}(u_0^{\ell-1}(0))x(0),u_0^{\ell-1}(1),\cdots,u_0^{\ell-1}(\ell-1)\right)\right]\\
&-\left[c(x(0),u_0^\infty(0))+J_1^{\ell-1}((u_0^\infty(0))x(0),u_0^\infty(1),\cdots,u_0^\infty(\ell-1))\right]\\
<& d.
\end{aligned} \tag{12.3.38}$$

由 (12.3.37) 及 (12.3.38) 可知

$$\left|EC(x(0),u_0^{\ell-1}(0))-EC(x(0),u_0^\infty(0))\right|<d. \tag{12.3.39}$$

由于所有控制都是确定性的, 并且 $u_0^{\ell-1}(0)\neq u_0^\infty(0)$, (12.3.39) 与 d 的定义矛盾. 因此可知

$$u^*(0)=u_0^{\ell-1}(0)=u_0^\infty(0). \tag{12.3.40}$$

同理可得

$$u^*(t) = u_t^{t+\ell-1}(t) = u_0^\infty(t), \quad t = 1, 2, \cdots. \tag{12.3.41}$$

□

注 12.3.2　条件 A-1 是很难检验的, 但在应用时可不检验. 只要预测控制得到稳态的最优反馈控制律, 它就是无穷时间最优控制律.

下面给出一个例子.

例 12.3.1　考虑一个人机博弈, 游戏为: 石头 (R)–剪刀 (S)–布 (P), 设 P_2 为机器, P_1 为人, 支付双矩阵见表 12.3.1.

表 12.3.1　石头-剪刀-布的支付双矩阵

P_1	P_2		
	R	S	P
R	0, 0	1, −1	−1, 1
S	−1, 1	0, 0	1, −1
P	1, −1	−1, 1	0, 0

设机器的策略如下:

S_1: 如果机器在 t 时刻赢了, 它将不改变策略. 否则它在 $t+1$ 时刻取对手 t 时刻的策略.

S_2: 机器依顺序 $R \to S \to P \to R$ 周期地变换策略.

S_3: 如果机器与人在 t 时刻策略相同, 机器在 $t+1$ 时刻不改变策略. 否则, 机器下一时刻取与 t 时刻未被取过的那个策略.

记 $R \sim \delta_3^1$, $S \sim \delta_3^2$, $P \sim \delta_3^3$, 并设 $x(t)$ 和 $u(t)$ 分别为机器和人在 t 时刻的策略. 那么, 对每个 S_i, $i = 1, 2, 3$, 机器的策略演化可表示为

$$x(t+1) = L_{S_i} u(t) x(t), \quad i = 1, 2, 3, \tag{12.3.42}$$

这里, 相应的结构矩阵可算出为

$$\begin{aligned} L_{S_1} &= \delta_3[1, 1, 3, 1, 2, 2, 3, 2, 3]; \\ L_{S_2} &= \delta_3[2, 3, 1, 2, 3, 1, 2, 3, 1]; \\ L_{S_3} &= \delta_3[1, 3, 2, 3, 2, 1, 2, 1, 3]. \end{aligned}$$

现在假定机器取 S_1 的概率为 0.4, 取 S_2 或 S_3 的概率均为 0.3, 那么, 演化方程可表示为

$$x(t+1) = L^* u(t) x(t), \tag{12.3.43}$$

这里,

$$
\begin{aligned}
L^* &= 0.4L_{S_1} + 0.3L_{S_2} + 0.3L_{S_3} \\
&= \begin{bmatrix} 0.7 & 0.4 & 0.3 & 0.4 & 0 & 0.6 & 0 & 0.3 & 0.3 \\ 0.3 & 0 & 0.3 & 0.3 & 0.7 & 0.4 & 0.6 & 0.4 & 0 \\ 0 & 0.6 & 0.4 & 0.3 & 0.3 & 0 & 0.4 & 0.3 & 0.7 \end{bmatrix}.
\end{aligned}
$$

设折扣系数 $\lambda = 0.9$. 根据支付双矩阵, 人的收益矩阵为

$$
\Phi = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix}. \tag{12.3.44}
$$

假定人有完全信息, 即, 机器的策略可知.

(i) 有限时间优化: 令 $T = N = 3$. 利用 (12.3.18)—(12.3.19), 可得

$$
\begin{aligned}
&u^*(3)\big|_{x(3)=\delta_3^1} = \delta_3^3, \quad J_3(\delta_3^1) = (0.9)^3; \\
&u^*(3)\big|_{x(3)=\delta_3^2} = \delta_3^1, \quad J_3(\delta_3^2) = (0.9)^3; \\
&u^*(3)\big|_{x(3)=\delta_3^3} = \delta_3^2, \quad J_3(\delta_3^3) = (0.9)^3.
\end{aligned}
$$

于是有

$$
J_3 = (0.9)^3 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}. \tag{12.3.45}
$$

利用 (12.3.25)—(12.3.26), 可算得

$$
\begin{aligned}
&u^*(2)\big|_{x(2)=\delta_3^1} = \delta_3^3, \quad J_2(\delta_3^1) = 1.539; \\
&u^*(2)\big|_{x(2)=\delta_3^2} = \delta_3^1, \quad J_2(\delta_3^2) = 1.539; \\
&u^*(2)\big|_{x(2)=\delta_3^3} = \delta_3^2, \quad J_2(\delta_3^3) = 1.539.
\end{aligned}
$$

继续这个过程, 可得

$$
\begin{aligned}
&u^*(1)\big|_{x(1)=\delta_3^1} = \delta_3^3, \quad J_1(\delta_3^1) = 2.439; \\
&u^*(1)\big|_{x(1)=\delta_3^2} = \delta_3^1, \quad J_1(\delta_3^2) = 2.439; \\
&u^*(1)\big|_{x(1)=\delta_3^3} = \delta_3^2, \quad J_1(\delta_3^3) = 2.439,
\end{aligned}
$$

以及

$$\begin{aligned}
u^*(0)\big|_{x(0)=\delta_3^1} &= \delta_3^3, \quad J_0(\delta_3^1) = 3.439;\\
u^*(0)\big|_{x(0)=\delta_3^2} &= \delta_3^1, \quad J_0(\delta_3^2) = 3.439;\\
u^*(0)\big|_{x(0)=\delta_3^3} &= \delta_3^2, \quad J_0(\delta_3^3) = 3.439.
\end{aligned}$$

(ii) 无穷时间优化: 不难看出

$$d = 1, \quad M = 1.$$

利用 (12.3.32) 可得

$$\ell > \log_{0.9} \frac{1-0.9}{2} = 28.4332.$$

不妨设 $\ell = 29$, 解 $T = 28$ 的有限时间优化问题可得

$$\begin{aligned}
u^*(0)\big|_{x(0)=\delta_3^1} &= \delta_3^3,\\
u^*(0)\big|_{x(0)=\delta_3^2} &= \delta_3^1,\\
u^*(0)\big|_{x(0)=\delta_3^3} &= \delta_3^2.
\end{aligned}$$

因此可知

$$u^*(t) = \left[u^*(0)\big|_{x(0)=\delta_3^1}, u^*(0)\big|_{x(0)=\delta_3^2}, u^*(0)\big|_{x(0)=\delta_3^3}\right] x(t) = \delta_3[3,1,2]x(t), \tag{12.3.46}$$

并且

$$J_0^\infty\big|_{x(0)=\delta_3^1} = J_0^\infty\big|_{x(0)=\delta_3^2} = J_0^\infty\big|_{x(0)=\delta_3^3} = 10.$$

参 考 文 献

[1] 程代展, 齐洪胜. 矩阵的半张量积——理论与应用. 2 版. 北京: 科学出版社, 2011.

[2] 程代展, 齐洪胜, 贺风华. 有限集上的映射与动态过程——矩阵半张量积方法. 北京: 科学出版社, 2016.

[3] 程代展, 夏元清, 马宏宾, 闫莉萍. 矩阵代数、控制与博弈. 北京: 北京理工大学出版社, 2016.

[4] 葛爱冬, 王玉振, 魏爱荣, 刘红波. 多变量模糊系统控制设计及其在并行混合电动汽车中的应用. 控制理论与应用, 2013, 30(8): 998-1004.

[5] 梅生伟, 刘锋, 薛安成. 电力系统暂态分析中的半张量积方法. 北京: 清华大学出版社, 2010.

[6] 欧阳城添, 江建慧. 基于概率转移矩阵的时序电路可靠度估计方法. 电子学报, 2013, 41(1): 171-177.

[7] 温巧燕, 钮心忻, 杨义先. 现代密码学中的布尔函数. 北京: 科学出版社, 2000.

[8] Crilly T. 你不可不知的 50 个数学知识. 王悦, 译. 北京: 人民邮电出版社, 2012.

[9] 金基恒. 布尔矩阵理论及其应用. 何善堉, 等, 译. 北京: 知识出版社, 1987.

[10] Ross S M. 应用随机过程: 概率模型导论. 龚光鲁, 译. 北京: 人民邮电出版社, 2011.

[11] Ashenhurst R L. The decomposition of switching functions. Annals Comp. Lab. Harvard Univ., 1957, 29: 74-116.

[12] Akutsu T, Hayashida M, Ching W, Ng M. Control of Boolean networks: Hardness results and algorithms for tree structured networks. J. Theor. Biol., 2007, 244(4): 670-679.

[13] Bazso F. Derivation of vector-valued Boolean functions. Acta Mathematica, Hungarica, 2000, 87(3): 197-203.

[14] Bellman R. Dynamic programming. Science, 1966, 153(3731): 34-37.

[15] Bertsekas D P. Dynamic Programming and Stochastic Control. New York: Academic Press, 1976.

[16] Brzezniak Z, Zastawniak T. Basic Stochastic Processes. New York: Springer-Verlag, 1999.

[17] Carlet C. Boolean Functions for Cryptography and Error Correctiong Codes. Cambridge: Cambridge University Press, 2010.

[18] Cheng D, Qi H. Controllability and observability of Boolean control networks. Automatica, 2009, 45(7): 1659-1667.

[19] Cheng D. Input-state approach to Boolean networks. IEEE Trans. Neural Networks, 2009, 20(3): 512-521.

[20] Cheng D, Li Z Q, Qi H. Realization of Boolean control networks. Automatica, 2010, 46(1): 62-69.

[21] Cheng D, Qi H. State-space analysis of Boolean networks. IEEE Trans. Neural Networks, 2010, 21(4): 584-594.

[22] Cheng D, Qi H, Li Z Q. Analysis and Control of Boolean Networks: A Semi-tensor Product Approach. London: Springer, 2011.

[23] Cheng D, Qi H, Li Z Q. Model construction of Boolean network via observed data. IEEE Trans. Neural Networks, 2011, 22(4): 525-536.

[24] Cheng D. Disturbance decoupling of Boolean control networks. IEEE Trans. Automatic Control, 2011, 56(1): 2-10.

[25] Cheng D, Qi H, Li Z Q, Liu J. Stability and stabilization of Boolean networks. Int. J. Robust Nonlinear and Control, 2011, 21(2): 134-156.

[26] Cheng D, Zhao Y. Identification of Boolean control networks. Automatica, 2011, 47(4): 702-710.

[27] Cheng D, Qi H, Zhao Y. An Introduction to Semi-Tensor Product of Matrices and Its Applications. Singapore: World Scientific, 2012.

[28] Cheng D, Feng J, Lv H. Solving fuzzy relational equations via semitensor product. IEEE Trans. Fuzzy Systems, 2012, 20(2): 390-396.

[29] Cheng D, Xu X. Bi-decomposition of multi-valued logical functions and its applications. Automatica, 2013, 49(7): 1979-1985.

[30] Cheng D, Xu T, Qi H. Evolutionarily stable strategy of networked evolutionary games. IEEE Trans. Neural Networks and Learning Systems, 2014, 25(7): 1335-1345.

[31] Cheng D. On finite potential games. Automatica, 2014, 50(7): 1793-1801.

[32] Cheng D, He F, Qi H, Xu T. Modeling, analysis and control of networked evolutionary games. IEEE Trans. Automatic Control, 2015, 60(9): 2402-2415.

[33] Cheng D, Zhao Y, Xu T. Receding horizon based feedback optimization for mix-valued logical networks. IEEE Trans. Automatic Control, 2015, 60(12): 3362-3366.

[34] Cheng D, Liu T, Zhang K, Qi H. On decomposed subspaces of finite games. IEEE Trans. Automatic Control, 2016, 61(11): 3651-3656.

[35] Chen H, Sun L J, Liu Y. Partial stability and stabilisation of Boolean networks. Int. J. Systems Science, 2016, 47(9): 2119-2127.

[36] Cheng D, Qi H, Liu T, Wang Y. A note on observability of Boolean control networks. Systems and Control Letters, 2016, 87: 76-82.

[37] Cheng D, Li C, He F. Observability of Boolean networks via set controllability approach. Systems and Control Letters, 2018, 155: 22-25.

[38] Cheng D. From Dimension-Free Matrix Theory to Cross-Dimensional Dynamic Systems. London: Academic Press, 2019.

[39] Datta A, Choudhary A, Bittner M, Dougherty E. External control in Markovian genetic regulatory networks. Machine Learning, 2003, 52: 169-191.

[40] Datta A, Choudhary A, Bittner M, Dougherty E. External control in Markovian genetic regulatory networks: The imperfect information case. Bioinformatics, 2004, 20: 924-930.

[41] Dixon J D, Mortimer B. Permutation Groups. London: Springer-Verlag, 1996.

[42] Farrow C, Heidel J, Maloney J. Scalar equations for synchronous Boolean networks with biological applications. IEEE Trans. Neural Networks, 2004, 15(2): 348-354.

[43] Feng J, Lv H, Cheng D. Multiple fuzzy relation and its application to coupled fuzzy control. Asian J. Control, 2013, 15(5): 1313-1324.

[44] Fornasini E, Valcher M E. Observability, reconstructibility and state observers of Boolean control networks. IEEE Trans. Automatic Control, 2013, 58: 1390-1401.

[45] Fornasini E, Valcher M E. Optimal control of Boolean control networks. IEEE Trans. Automatic Control, 2014, 59(5): 1258-1270.

[46] Fornasini E, Valcher M E. Identification problems for Boolean networks and Boolean control networks. Proc. 19th IFAC, Cape Town, August 24-29, 2014.

[47] Fornasini E, Valcher M E. Recent developments in Boolean networks control. J. Control Decision, 2016, 3(1): 1-18.

[48] Fornasini E, Valcher M E. Observability and reconstructibility of probabilistic Boolean networks. IEEE Control Systems Letters, 2020, 4(2): 319-324.

[49] Fu S, Wang Y, Cheng D, Liu J. Morgan's problem of Boolean control networks. Control Theory Technology, 2017, 15(4): 316-326.

[50] Gao B, Li L, Peng H, Kurths J, Zhang W, Yang Y. Principle for performing attractor transits with single control in Boolean networks. Physical Review E, 2013, 88(6): 062706.

[51] Gao B, Peng H, Zhao D, Zhang W, Yang Y. Attractor transformation by impulsive control in Boolean control network. Mathematical Problems in Engineering, 2013, 2013: 674571.

[52] Gibbons R. A Primer in Game Theory. New York: Printice Hall, 1992.

[53] Glumineau A, Moog C H. Nonlinear Morgan's problem: Case of $(p+1)$ inputs and p outputs. IEEE Trans. Automatic Control, 1992, 37(7): 1067-1072.

[54] Goodwin B. Temporal Organization in Cells. San Diego: Academic Press, 1963.

[55] Guo P, Wang Y, Li H. Algebraic formulation and strategy optimization for a class of evolutionary networked games via semi-tensor product method. Automatica, 2013, 49(11): 3384-3389.

[56] Guo Y, Wang P, Gui W, Yang C. Set stability and set stabilization of Boolean control networks based on invariant subsets. Automatica, 2015, 61: 106-112.

[57] Guo Y, Zhou R, Wu Y, Gui W, Yang C. Stability and set stability in distribution of probabilistic Boolean networks. IEEE Trans. Automatic Control, 2019, 64(2): 736-742.

[58] Hachtel G D, Somenzi F. Logic Synthesis and Verification Algorithms. Boston: Kluwer Academic Public., 2002.

[59] Hao Y, Cheng D. On skew-symmetric games. J. Franklin Institute, 2018, 355: 3196-3220.

[60] Heidel J, Maloney J, Farrow C, Rogers J. Finding cycles in synchronous Boolean networks with applications to biochemical systems. Int. J. Bifurcation and Chaos, 2003, 13(3): 535-552.

[61] Hochma G, Margaliot M, Fornasini E. Symbolic dynamics of Boolean control networks. Automatica, 2013, 49(8): 2525-2530.

[62] Horn R A, Johnson C R. Matrix Analysis. Cambridge: Cambridge University. Press, 1986.

[63] Huang S, Ingber D. Shape-dependent control of cell growth, differentiation and apoptosis: Switching between attractors in cell regulatory networks. Exp. Cell Res., 2000, 261(1): 91-103.

[64] Huang J. Nonlinear Output Regulation: Theory and Applications. Philadelphia: SIAM, 2004.

[65] Ideker T, Galitski T, Hood L. A new approach to decoding life: Systems biology. Annu. Rev. Genomics Hum. Genet., 2001, 2: 343-372.

[66] Isidori A. Nonlinear Control Systems. 3rd ed. Berlin: Springer, 1995.

[67] Jia Y, Cheng D, Feng J. State feedback stabilization of generic logic systems via Ledley antecedence solution. Mathematical Methods in the Applied Sciences, 2021, 1-11.

[68] Jarrah A S, Raposa B, Laubenbacher R. Nested canalizing, unite cascade, and polynomial functions. Physica D, 2007, 233: 167-174.

[69] Johnson C D. Accomodation of external disturbances in linear regulator and servomechanism problems. IEEE Trans. Automatic Control, 1971, 16: 635-644.

[70] Just W, Shmulevich I, Konvalina H. The number and probability of canalizing functions. Physica D, 2004, 197: 211-221.

[71] Kalman R E. Canonical structure of linear dynamical systems. Proc. Nat. Acad. Sci. US, 1962, 48: 596-600.

[72] Katz V J. A History of Mathematics: Brief Version. New York: Addison-Wesley, 2004.

[73] Kauffman S A. The Origins of Order: Self-Organization and Selection in Evolution. New York: Oxford University Press, 1993.

[74] Kauffman S A. At Home in the Universe. Oxford: Oxford University. Press, 1995.

[75] Kauffman S A, Peterson C, Samuelsson B, Troein C. Random Boolean network models and the yeast transcriptional network. Proc. Natl. Acad. Sci. USA, 2004, 100(25): 14796-14799.

[76] Kauffman S A, Peterson C, Samuelsson B, Troein C. Genetic Networks with Canalizing Boolean rules are always stable. Proc. Natl. Acad. Sci. USA, 2004, 101(49): 17102-17107.

[77] Kwon W H, Han S. Receding Horizon Control: Model Predictive Control for State Models. London: Springer-Verlag, 2005.

[78] Lancaster P, Tismenetsky M. The Theory of Matrices: With Applications. London: Elsevier, 1985.

[79] Lang C, Steinbach B. Bi-decomposition of function sets in multiple-valued logic for circuit design and data mining. Artificial Intelligence Review, 2003, 20: 233-267.

[80] Laschov D, Margaliot M, Even G. Observability of Boolean networks: A graph-theoretic approach. Automatica, 2013, 49: 2351-2362.

[81] Laschov D, Margaliot M. Minimum-time control of Boolean networks. SIAM J. Control and Optimization, 2013, 51(4): 2869-2892.

[82] Ledley R S. Digital computational methods in symbolic logic with examples in biochemistry. Proc. Nat. Acad. Sci. USA, 1955, 41(7): 498-511.

[83] Ledley R S. Logic and Boolean algebra in medical science. Proc. of Conference on Application of Undergraduate Mathematics, Atlanta, GA, 1973.

[84] Li Z, Cheng D. The structure of canalizing functions. J. Control Theory Applications, 2010, 8(3): 375-381.

[85] Li Z, Zhao Y, Cheng D. Structure of higher order Boolean networks. J. Grad. Univ. C.A.S, 2011, 28(4): 431-447.

[86] Li F, Sun J. Controllability of probabilistic Boolean control networks. Automatica, 2011, 47(12): 2765-2771.

[87] Li H, Wang Y. Boolean derivative calculation with application to fault detection of combinational circuits via the semi-tensor product method. Automatica, 2012, 48(4): 688-693.

[88] Li R, Yang M, Chu T. State feedback stabilization for Boolean control networks. IEEE Trans. Automatic Control, 2013, 58(7): 1853-1857.

[89] Li H, Wang Y. Output feedback stabilization control design for Boolean control networks. Automatica, 2013, 49(12): 3641-3645.

[90] Li R, Yang M, Chu T. State feedback stabilization for probabilistic Boolean networks. Automatica, 2014, 50(4): 1272-1278.

[91] Li H, Wang Y, Xie L. Output tracking control of Boolean control networks. Automatica, 2015, 59: 54-59.

[92] Li R, Yang M, Chu T. State feedback stabilization for Boolean control networks. IEEE Trans. Automatic Control, 2017, 58: 1853-1857.

[93] Li H, Zhao G, Guo P, Liu Z. Analysis and Control of Finite-Valued Systems. Boca Raton: CRC Press, 2018.

[94] Li H, Yang X, Wang S. Robustness for stability and stabilization of Boolean networks with stochastic function perturbations. IEEE Trans. Automatic Control, 2020, 66(3): 1231-1237.

[95] Li H, Xie L, Wang Y. Output regulation of Boolean control networks. IEEE Trans. Automatic Control, 2017, 62(6): 2993-2998.

[96] Liu Z, Wang Y, Li H. New approach to derivative calculation of multi-valued logical functions with application to fault detection of digital circuits. IET Control Theory & Applications, 2014, 8(8): 554-560.

[97] Liu Z, Wang Y, Cheng D. Nonsingularity of nonlinear feedback shift registers. Automatica, 2015, 55: 247-253.

[98] Liu Z, Cheng D. Canonical form of Boolean networks. Proc. 38th CCC, 2019: 1801-1806.

[99] Lu J, Li H, Liu Y, Li F. Survey on semi-tensor product method with its applications in logical networks and other finite-valued systems. IET Control Theory & Applications, 2017, 11(13): 2040-2047.

[100] Meng M, Feng J. A matrix approach to hypergraph stable set and coloring problems with its application to storing problem. Journal of Applied Mathematics, 2014, 2014: 783784.

[101] Meng M, Lam J, Feng J, Cheung K C. Stability and stabilization of Boolean networks with stochastic delays. IEEE Trans. Automatic Control, 2019, 64(2): 790-796.

[102] Morgan B S. The synthesis of linear multivariable systems by state-variable feedback. IEEE Trans. Automatic Control, 1964, 9(4): 405-411.

[103] Mu Y, Guo L. Optimization and identification in a non-equilibrium dynamic game. Proc. CDC-CCC'09, 2009: 5750-5755.

[104] Muhammad A, Rushdi A, Ghaleb F A M. A tutorial exposition of semi-tensor products of matrices with a stress on their representation of Boolean functions. JKAU Comp. Sci., 2016, 5: 3-30.

[105] Nijmeijer H. Feedback decomposition of nonlinear control systems. IEEE Trans. Automatic Control, 1983, 28(8): 861-862.

[106] Ostrowski M, Paulevé L, Schaub T, Siegel A, Guziolowski C. Boolean network identification from perturbation time series data combining dynamics abstraction and logic programming. Biosystems, 2016, 149: 139-153.

[107] Pal R, Datta A, Dougherty E R. Optimal infinite-horizon control for probabilistic Boolean networks. IEEE Trans. Signal Processing, 2006, 54(6): 2375-2387.

[108] Qiao Y, Qi H, Cheng D. Partition-based solutions of static logical networks with applications. IEEE Trans. Neural Networks and Learning Systems, 2018, 29(4): 1252-1262.

[109] Qi H, Qiao Y. Dynamics and control of singular Boolean networks. Asia J. Control, 2019, 21(6): 2604-2613.

[110] Robert F. Discrete Iterations: A Metric Study. Berlin: Springer-Verlag, 1986.

[111] Robeva R, Hodge T. Mathematical Concepts and Methods in Modern Biology: Using Modern Discrete Models. New York: Academic Press, 2013.

[112] Shmulevich I, Dougherty E, Kim S, Zhang W. Probabilistic Boolean networks: A rule-based uncertainty model for gene regulatory networks. Bioinformatics, 2002, 18(2): 261-274.

[113] Shmulevich I, Lähdesmäki H, Egiazarian K. Special methods for testing membership in certain post classes and class of forcing functions. IEEE Signal Processing Letters, 2004, 11(2): 289-292.

[114] Veliz-Cuba A, Stigler B. Boolean models can explain bistability in the lac operon. J. Comput. Biol., 2011, 18(6): 783-794.

[115] Vykhovanets V S. Algebraic decomposition of discrete functions. Automation and Remote Control, 2016, 67(3): 361-392.

[116] Waldrop M M. Complexity: The Emerging Science at the Edge of Order and Chaos. New York: Touchstone, 1992.

[117] Wang Y, Zhang C, Liu Z. A matrix approach to graph maximum stable set and coloring problems with application to multi-agent systems. Automatica, 2012, 48(7): 1227-1236.

[118] Wang B, Feng J, Cheng D. On identification of Boolean control networks. 2021. arXiv: 2104.14144.

[119] Wolfram S. Theory and Applications of Cellular Automata. Singapore: World Scientific, 1986.

[120] Wonham W. Linear Multivariable Control: A Geometric Approach. 2nd ed. Berlin: Springer-Verlag, 1979.

[121] Wu Y, Shen T. An algebraic expression of finite horizon optimal control algorithm for stochastic logical dynamical systems. Systems & Control Letters, 2015, 82: 108-114.

[122] Wu Y, Kumar M, Shen T. A stochastic logical system approach to model and optimal control of cyclic variation of residual gas fraction in combustion engines. Appl. Therm. Eng., 2016, 93: 251-259.

[123] Wu Y, Shen T. A finite convergence criterion for the discounted optimal control of stochastic logical networks. IEEE Trans. Automatic Control, 2018, 63(1): 262-268.

[124] Hua X, Duan P, Lv H, Zhang Z. Design of fuzzy controller for air-conditioning systems based-on Semi-tensor Product. Proc. 26th Chinese Control and Decision Conference, Changsha, 2014: 3507-3512.

[125] Xu X, Hong Y. Matrix expression and reachability analysis of finite automata. J. Control Theory and Applications, 2012, 10(2): 210-215.

[126] Xu X, Hong Y. Matrix approach to model matching of asynchronous sequential machines. IEEE Trans. Automatic Control, 2013, 58(11): 2974-2979.

[127] Xu X, Hong Y. Observability analysis and observer design for finite automata via matrix approach. IET Control Theory & Applications, 2013, 7(12): 1609-1615.

[128] Xu M, Wang Y, Wei A. Robust graph coloring based on the matrix semi-tensor product with application to examination timetabling. Control Theory and Technology, 2014, 12(2): 187-197.

[129] Yan Y, Chen Z, Liu Z. Solving type-2 fuzzy relation equations via semi-tensor product of matrices. Control Theory and Technology, 2014, 12(2): 173-186.

[130] Yan Y, Chen Z, Liu Z. Semi-tensor product of matrices approach to reachability of finite automata with application to language recognition. Front. Comput. Sci., 2014, 8(6): 948-957.

[131] Yan Y, Chen Z, Liu Z. Semi-tensor product approach to controllability and stabilizability of finite automata. J. Syst. Engn. Electron., 2015, 26(1): 134-141.

[132] Yu Y, Feng J, Pan J, Cheng D. Block decoupling of Boolean control networks. IEEE Trans. Automatic Control, 2019, 64(8): 3129-3140.

[133] Zhang K, Zhang L, Xie L. Discrete-time and Discrete-Space Dynamical Systems. Berlin: Springer, 2020.

[134] Zhan L, Feng J. Mix-valued logic-based formation control. Int. J. Control, 2013, 86(6): 1191-1199.

[135] Zhao Q. A remark on "Scalar equations for synchronous Boolean networks biological applications" by C. Farrow, J. Maloney, and J. Rogers. IEEE Trans. Neural Networks, 2005, 16(6): 1715-1716.

[136] Zhao Y, Qi H, Cheng D. Input-state incidence matrix of Boolean control networks and its applications. Systems & Control Letters, 2010, 59(12): 767-774.

[137] Zhao Y, Li Z, Cheng D. Optimal control of logical control networks. IEEE Trans. Automatic Control, 2011, 56(8): 1766-1776.

[138] Zhan J, Lu S, Yang G. Improved calculation scheme of structure matrix of Boolean network using semi-tensor product// Liu C, Wang L, Yang A, eds. Information Computing and Applications, Part 1, Book Series: Communications in Computer and Information Science. Berlin, Heidelberg: Springer, 2012, 307: 242-248.

[139] Zhao D, Peng H, Li L, Sili H, Yang Y. Novel way to research nonlinear feedback shift register. Science China, Information Sciences, 2014, 57(9): 1-14.

[140] Zhao Y, Cheng D. On controllability and stabilizability of probabilistic Boolean control networks. Science China, Information Sciences, 2014, 57(1): 1-14.

[141] Zhao J, Liu Z. Observability of probabilistic Boolean networks. Proc. CCC2015, 2015: 183-186.

[142] Zhao Y, Kim J, Filippone M. Aggregation algorithm towards large-scale Boolean network analysis. IEEE Trans. Automatic Control, 2013, 58(8): 1976-1985.

[143] Zhao Y, Ghosh B, Cheng D. Control of large-scale Boolean networks via network aggregation. IEEE Trans. Neural Networks and Learning Systems, 2016, 27(7): 1527-1536.

[144] Zhang K, Zhang L. Observability of Boolean control networks: A unified approach based on finite automata. IEEE Trans. Automatic Control, 2016, 61(9): 2733-2738.

[145] Zhang K, Liu T, Cheng D. Observability of finite labeled transition systems. IEEE Trans. Automatic Control, 2018, 63(6): 1591-1602.

[146] Zhang X, Wang Y, Cheng D. Output tracking of Boolean control networks. IEEE Trans. Automatic Control, 2020, 65(6): 2730-2735.

[147] Zhong J, Lin D. A new linearization method for nonlinear feedback shift registers. Journal of Computer and System Sciences, 2015, 81: 783-796.

[148] Zhong J, Lin D. Decomposition of nonlinear feedback shift registers based on Boolean networks. Science China Information Sciences, 2019, 62(3): 039110.

[149] Zhong J, Lin D. Stability of nonlinear feedback shift registers. Science China Information Sciences, 2016, 59(1): 1-12.

[150] Zhong J, et al. Global robust stability and stabilization of Boolean network with disturbances. Automatica, 2017, 84: 142-148.

[151] Zhou R, Guo Y, Gui W. Set reachability and observability of probabilistic Boolean networks. Automatica, 106: 230-241, 2019.

[152] Zhou R, Guo Y, Wu Y, Gui W. Asymptotical feedback set stabilization of probabilistic Boolean control networks. IEEE Trans. Neural Networks and Learning Systems, 2020, 31(11): 4524-4537.

[153] Zhu S, Lu J, Ho D W C. Finite-time stability of probabilistic logical networks: A topological sorting approach. IEEE Trans. Circ. Sys. - II: Express Briefs, 2020, 67(4): 695-699.

索　引

(O-8542.31)

LECTURE NOTES IN SEMI-TENSOR PRODUCT OF MATRICES

ANALYSIS AND CONTROL OF LOGICAL DYNAMIC SYSTEMS

矩阵半张量积讲义



逻辑动态系统的分析与控制

www.sciencep.com



科学出版社互联网入口
科学数理分社：(010) 64011058　销售：(010) 64031535
E-mail：lixin_kx@mail.sciencep.com
销售分类建议：高等数学



定 价：188.00 元